Programming Manual

VectorStar™ MS464xB Series Microwave Vector Network Analyzer

MS4642B VNA, 10 MHz to 20 GHz, K Connectors

MS4644B VNA, 10 MHz to 40 GHz, K Connectors

MS4645B VNA, 10 MHz to 50 GHz, V Connectors

MS4647B VNA, 10 MHz to 70 GHz, V Connectors

Anritsu

Anritsu Company
490 Jarvis Drive
Morgan Hill, CA 95037-2809
USA
http://www.anritsu.com

Part Number: 10410-00322
Version 2.3.2 Revision: L
Published: September 2016

Table of Contents

Chapter 1—General Information

1-1 About this Manual . . . 1-1

1-2 Introduction . . . 1-1

1-3 Related Documentation . . . 1-2
- Product Information, Compliance, and Safety . . . 1-2

1-4 Contacting Anritsu . . . 1-2

1-5 IEEE 488 GPIB Description . . . 1-3
- GPIB Network Restrictions . . . 1-4
- Functional Elements . . . 1-6
- Bus Structure . . . 1-7
- Data Bus Description . . . 1-7
- Data Byte Transfer Control Bus Description . . . 1-8
- General Interface Management Bus Description . . . 1-8
- Device Interface Function Capability . . . 1-9
- Message Types . . . 1-10
- Interface Function Messages . . . 1-10
- Device-Specific Commands . . . 1-10
- Data and Instrument Status Messages . . . 1-10
- Response to GPIB Interface Function Messages . . . 1-11
- Configuring the Dedicated GPIB Port . . . 1-12
- Configuring the IEEE 488.2 GPIB Port . . . 1-16
- To change the VectorStar VNA GPIB address: . . . 1-17

1-6 Resetting GPIB Addresses . . . 1-17
- Background . . . 1-17
- Returning the GPIB Addresses to Default . . . 1-17

1-7 Ethernet LAN TCP/IP and USB Description . . . 1-18
- TCP/IP General Requirements and Settings . . . 1-18
- Default Plug-and-Play Configuration . . . 1-19
- Manually Configuring TCP/IP Ethernet LAN Settings . . . 1-20

1-8 Configuring the Remote Language . . . 1-21
- Procedure . . . 1-21

1-9 Minimum/Maximum Instrument Frequency and Related Parameters . . . 1-22
- Standalone VNAs . . . 1-22
- Standalone VNAs – Default Start, Default CW, and Default Stop Frequencies . . . 1-23
- Standalone VNAs – Minimum Start, Minimum CW, and Maximum Start Frequencies . . . 1-24
- Standalone VNAs – Default Frequency Span and Maximum Frequency Span . . . 1-26
- Standalone VNAs – Minimum Center and Maximum Center Frequencies . . . 1-27
- Standalone VNAs – Default Center Frequencies . . . 1-28
- VNA Systems with ME7828A Configured as Broadband System . . . 1-28
- VNA Systems with ME7828A Configured as a Millimeter-Wave System . . . 1-28
- VNA Systems with ME7828A Using Multiple Source . . . 1-28
- VNA Systems with ME7838x Modular Broadband System . . . 1-28
- VNA Systems with ME7838x Configured as a Millimeter-Wave System . . . 1-28

Chapter 2—Programming the VectorStar Series VNA

2-1 Introduction . . . 2-1

2-2 Introduction to SCPI Programming . . . 2-1
- Command Types . . . 2-1

2-3 IEEE 488.2 Commands . . . 2-2

2-4 System Commands. . . . 2-2

2-5 SCPI Commands . . . 2-2
- Required SCPI Commands . . . 2-2
- Native SCPI Commands . . . 2-3

2-6 Anritsu Lightning VNA Commands . . . 2-3
- Using Anritsu Lightning VNA Commands . . . 2-3
- VectorStar VNA Language Set to Lightning . . . 2-3

2-7 HP8510 Commands . . . 2-4

2-8 Command Requirements . . . 2-5
- Query Commands . . . 2-5
- Command Names . . . 2-5
- Hierarchical Command Structure. . . . 2-5
- Data Parameters . . . 2-6

2-9 Notational Conventions. . . . 2-7
- General Notations . . . 2-7
- Parameter Notations . . . 2-8
- Notational Examples . . . 2-9
- Band Equation . . . 2-9

2-10 Numeric Data Suffix Reference . . . 2-10

2-11 Data Transmission Methods . . . 2-11
- <NR1> . . . 2-11
- <NR2> . . . 2-11
- <NR3> . . . 2-11
- <NRf> . . . 2-11
- <string> . . . 2-12
- <ASCII> or <Arbitrary ASCII> . . . 2-12
- <block> or <arbitrary block> . . . 2-12
- <char> . . . 2-14
- MPND . . . 2-14
- MPNF. . . . 2-14
- MPNI . . . 2-14
- Formatting Data Output . . . 2-14
- ASCII or Binary Data Format . . . 2-15

2-12 Calculating the Byte Size . . . 2-18
- Numbers Output-per-Data Point (NODP) . . . 2-18
- Bytes Output-per-Number (BOPN) . . . 2-18
- Size of Data Block (SODB) . . . 2-19
- Number of Bytes Output (NBO) . . . 2-19

2-13 GPIB Input Buffer Size and NRFD Holdoff . . . 2-19

2-14 Synchronization of GPIB Commands 2-20
2-15 Forcing the Parser to Stop Waiting. 2-20
2-16 How Can I Abort an RF or Hardware Calibration 2-20
2-17 GPIB Time-Out Settings 2-20
2-18 Trace Type Parameters and Coefficients 2-22
2-19 Input/Output Data Files. 2-26
2-20 Status System Reporting 2-29
Status Group Registers 2-29
Status Group Reporting. 2-29
2-21 Trigger System 2-35
Trigger Modes 2-35
Trigger Timing 2-37
2-22 Calibration Component Parameters 2-38
Loads and Through Lines 2-38
Other Connector Coefficients 2-39
W1 Calibration Kit 2-40
Sliding Load Cutoff Frequency 2-41
Microstrip Kit Common Values 2-42
2-23 Notes on Calibration Commands 2-44
Setting Up a Two-Port Calibration 2-44
Setting Up a Four-Port Calibration 2-45
Defining the Calibration Standards 2-48
Performing the Calibration. 2-52
AutoCal 2-54
LRL Calibration 2-59
2-24 Command Script Example – Limit Lines. 2-62
Limit Lines for Single Rectilinear Trace Display 2-62
Required Equipment 2-63
Prerequisites 2-63
DUT Requirements 2-63
Channel and Trace Display Requirements 2-63
VNA General Setup and Configuration 2-64
Frequency and Sweep Settings. 2-65
Limit Lines Setup 2-65
Clear Previous Limit Lines. 2-66
Create and Configure Limit Line Segment 1 2-66
Create and Configure Limit Line Segment 2 2-66
Create and Configure Limit Line Segment 3 2-67
Create and Configure Limit Line Segment 4 2-67
Configure AutoCal Calibration 2-68
Ready for Measurements 2-68

Chapter 3—IEEE Commands

3-1 Introduction. 3-1
3-2 Command Descriptions 3-1

3-3 Numeric Limits 3-2
3-4 IEEE 488.2 Commands 3-2

Chapter 4—System and Troubleshooting Commands

4-1 Introduction 4-1
4-2 Parameters and Notations 4-1
4-3 Numeric Limits 4-2
4-4 System Commands. 4-3

Chapter 5—SCPI Commands

5-1 Introduction 5-1
5-2 Minimum/Maximum Frequency Limits and Related Parameters 5-1
5-3 Command Level Hierarchy 5-2
Command Descriptions and Notation Conventions 5-2
Numeric Limits 5-2
Notational Conventions 5-3
5-4 General Parameters 5-4
5-5 :CALCulate{1-16}:AFCW Subsystem 5-6
5-6 :CALCulate{1-16}:APPLication:MEASurement Subsystem 5-13
5-7 :CALCulate{1-16}:CORRection Subsystem - Adapters/Merge Calibration 5-14
Calibration Option Subsystems 5-14
5-8 :CALCulate{1-16}:DISPlay:MARKer Subsystem. 5-17
Marker Subsystems 5-17
5-9 :CALCulate{1-16}:EOOE: Subsystem. 5-18
5-10 :CALCulate{1-16}:EXTRaction Subsystem - Network Extraction 5-33
Calibration Option Subsystems 5-33
General Parameters 5-33
5-11 :CALCulate{1-16}:FCW Subsystem 5-44
5-12 :CALCulate{1-16}:FORMat Subsystem - SnP Data 5-49
I/O Configuration and File Operation Subsystems 5-49
5-13 :CALCulate{1-16}:FSIMulator:NETWork Subsystem - Simulation 5-51
Calibration Simulation Subsystems 5-51
5-14 :CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation 5-63
Calibration Simulation Subsystems 5-63
5-15 :CALCulate{1-16}:IMPedance:TRANsformation Subsystem 5-74
Calibration Setup Subsystems. 5-74
5-16 :CALCulate{1-16}:MARKer Subsystem. 5-76
Marker Subsystems 5-76
5-17 :CALCulate{1-16}:MXP Subsystem 5-77
Measurement and Mixed Mode Subsystems. 5-77
General Parameters 5-77
5-18 :CALCulate{1-16}:NFIGure Subsystem 5-83

5-19 :CALCulate{1-16}:NXN Subsystem 5-88
Calibration Option Subsystems 5-88
I/O Configuration and File Operation Subsystems 5-88
5-20 :CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem 5-92
Trace Subsystems. 5-92
5-21 :CALCulate{1-16}:PARameter{1-16}:REFerece Subsystem. 5-99
5-22 :CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem 5-101
Measurement and Mixed Mode Subsystems 5-101
5-23 :CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem 5-107
Marker Subsystems. 5-107
5-24 :CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem 5-110
Marker Subsystems. 5-110
5-25 :CALCulate{1-16}:PARameter{1-16}:SELect Subsystem 5-112
Trace Subsystems. 5-112
5-26 :CALCulate{1-16}:POLar Subsystem 5-113
Trace Subsystems. 5-113
5-27 :CALCulate{1-16}:PROCessing:ORDer Subsystem 5-115
Time Domain, Group Delay, and Reference Plane Subsystems. 5-115
5-28 :CALCulate{1-16}:PULSe Subsystem. 5-116
Pulse Configuration and Setup Validation 5-116
5-29 :CALCulate{1-16}:REFerence Subsystem 5-127
Calibration Setup Subsystems 5-127
Time Domain, Group Delay, and Reference Plane Subsystems. 5-127
5-30 :CALCulate{1-16}[:SELected]:CONVersion Subsystem 5-134
Trace Subsystems. 5-134
5-31 :CALCulate{1-16}[:SELected]:DATA Subsystem 5-136
I/O Configuration and File Operation Subsystems 5-136
Trace Subsystems. 5-136
5-32 :CALCulate{1-16}[:SELected]:EYE Subsystem 5-138
5-33 :CALCulate{1-16}[:SELected]:FORmat Subsystem 5-147
Trace Subsystems. 5-147
5-34 :CALCulate{1-16}[:SELected]:GCOMpression Subsystem. 5-149
Time Domain, Group Delay, and Reference Plane Subsystems. 5-149
5-35 :CALCulate{1-16}[:SELected]:GDELay Subsystem 5-152
Time Domain, Group Delay, and Reference Plane Subsystems. 5-152
5-36 :CALCulate{1-16}[:SELected]:LIMit Subsystem 5-153
Limit Line and Segment Subsystems 5-153
Trace Subsystems. 5-153
5-37 :CALCulate{1-16}[:SELected]:MARKer Subsystem 5-165
Marker Subsystems. 5-165
5-38 :CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem 5-178
Marker Subsystems. 5-178
Trace Subsystems. 5-178

5-39 :CALCulate{1-16}[:SELected]:MATH Subsystem 5-181
Trace Subsystems 5-181
5-40 :CALCulate{1-16}[:SELected]:MDATA Subsystem 5-185
Trace Subsystems 5-185
5-41 :CALCulate{1-16}[:SELected]:MSTatistics Subsystem 5-187
5-42 :CALCulate{1-16}[:SELected]:SMITh Subsystem 5-194
Trace Subsystems 5-194
5-43 :CALCulate{1-16}[:SELected]:SMOothing Subsystem 5-196
Trace Subsystems 5-196
5-44 :CALCulate{1-16}[:SELected]:TDATA Subsystem 5-197
Trace Subsystems 5-197
I/O Configuration and File Operation Subsystems 5-197
5-45 :CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem 5-199
Time Domain, Group Delay, and Reference Plane Subsystems 5-199
Trace Subsystems 5-199
5-46 :CALCulate{1-16}:SNPSetup Subsystem 5-212
5-47 :CALCulate{1-16}:TMS Subsystem. 5-213
5-48 :CONTrol:AUXio Subsystem. 5-215
Rear Panel Subsystems 5-215
5-49 :CONTrol{1-16}:AOUT Subsystem 5-216
Rear Panel Subsystems 5-216
I/O Configuration and File Operation Subsystems 5-216
Setting the Rear Panel Output Mode as Horizontal 5-216
Setting the Rear Panel Output Mode as Driven Port 5-217
Setting the Rear Panel Output Mode as TTL. 5-217
5-50 :DISPlay Subsystem 5-223
Trace Subsystems 5-223
Marker Subsystems 5-223
Limit Line Subsystems 5-223
5-51 :FORMat Subsystem. 5-242
I/O Configuration and File Operation Subsystems 5-242
5-52 :HCOPy Subsystem 5-245
I/O Configuration and File Operation Subsystems 5-245
5-53 :MMEMory Subsystem 5-251
I/O Configuration and File Operation Subsystems 5-251
5-54 :SENSe:BANDwidth Subsystem 5-260
IF Configuration Subsystems. 5-260
5-55 :SENSe:HOLD Subsystem 5-261
Trigger, Hold, and External Source Subsystems 5-261
5-56 :SENSe{1-16}:ABORtcal Subsystem 5-264
Calibration Subsystems with Actual Calibrations. 5-264
5-57 :SENSe{1-16}:AVERage Subsystem 5-265
Channel and Sweep Subsystems 5-265

5-58 :SENSe{1-16}:BANDwidth Subsystem 5-267
IF Configuration Subsystems 5-267
5-59 :SENSe{1-16}:BWIDth Subsystem 5-268
IF Configuration Subsystems 5-268
5-60 :SENSe{1-16}:CORRection:COEFficient:PORT Subsystem - Simulation 5-269
Calibration Simulation Subsystems 5-269
Calibration Type Abbreviations 5-269
Related Query 5-269
5-61 :SENSe{1-16}:CORRection:COEFficient Subsystem - Simulation 5-274
Calibration Simulation Subsystems 5-274
Calibration Type Abbreviations 5-274
5-62 :SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs. 5-279
Calibration Subsystems with Actual Calibration 5-279
Calibration Type Abbreviations 5-279
Related Query 5-279
5-63 :SENSe{1-16}:CORRection:COLLect:ECAL Subsystem - 4-Port VNAs. 5-281
5-64 :SENSe{1-16}:CORRection:COLLect:DISPlay Subsystem 5-287
5-65 :SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem. 5-288
Calibration Option Subsystems 5-288
5-66 :SENSe{1-16}:CORRection:COLLect:FULL4 Subsystem - 4-Port VNAs 5-289
Calibration Subsystems with Actual Calibration 5-289
5-67 :SENSe{1-16}:CORRection:COLLect:HYBRid Subsystem 5-290
Calibration Option Subsystems 5-290
Calibration Type Abbreviations 5-290
5-68 :SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem 5-295
LRL Calibration Subsystems 5-295
5-69 :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem 5-305
LRL Calibration Subsystems 5-305
5-70 :SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs 5-309
LRL Calibration Subsystems 5-309
5-71 :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs 5-311
LRL Calibration Subsystems 5-311
5-72 :SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide Subsystem. 5-316
LRL Calibration Subsystems 5-316
5-73 :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem. 5-317
LRL Calibration Subsystems 5-317
5-74 :SENSe{1-16}:CORRection:COLLect:METHod Subsystem 5-329
Calibration Setup Subsystems 5-329
5-75 :SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem. 5-330
Calibration Setup Subsystems 5-330
5-76 :SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem 5-335
Calibration Setup Subsystems 5-335

5-77 :SENSe{1-16}:CORRection:COLLect:PORT Subsystem 5-336
Calibration Subsystems with Actual Calibrations. 5-336
Calibration Type Abbreviations 5-336
5-78 :SENSe{1-16}:CORRection:COLLect Subsystem 5-371
Calibration Setup Subsystems. 5-371
5-79 :SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem 5-373
Calibration Setup Subsystems. 5-373
5-80 :SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem 5-380
Calibration Subsystems with Actual Calibration 5-380
Calibration Abbreviations. 5-380
5-81 :SENSe{1-16}:CORRection:COLLect Subsystem - 2-Port/4-Port VNA 5-382
Calibration Setup Subsystems. 5-382
Related Command Subsystems 5-382
Calibration Method Names 5-382
5-82 :SENSe{1-16}:CORRection:DISPlay Subsystem - 2-Port VNA 5-394
5-83 :SENSe{1-16}:CORRection:EXTension Subsystem. 5-395
Time Domain, Group Delay, and Reference Plane Subsystems. 5-395
5-84 :SENSe{1-16}:CORRection:INTerpolation Subsystem. 5-396
Calibration Setup Subsystems. 5-396
5-85 :SENSe{1-16}:CORRection:ISOLation Subsystem. 5-397
Calibration Setup Subsystems. 5-397
5-86 :SENSe{1-16}:CORRection:STATe Subsystem 5-398
Calibration Setup Subsystems. 5-398
5-87 :SENSe{1-16}:FREQuency Subsystem 5-399
Channel and Sweep Subsystems 5-399
Frequency Limits 5-399
5-88 :SENSe{1-16}:FSEGMent Subsystem 5-401
Limit Line and Segment Subsystems 5-401
5-89 :SENSe{1-16}:FSEGMent{1-50} Subsystem 5-412
Limit Line and Segment Subsystems 5-412
5-90 :SENSe{1-16}:GRANging Subsystem. 5-418
5-91 :SENSe{1-16}:HOLD Subsystem 5-419
Trigger, Hold, and External Source Subsystems. 5-419
5-92 :SENSe{1-16}:IMD Subsystem 5-420
IMD Configuration and Setup Validation Steps 5-420
5-93 :SENSe{1-16}:ISEGMent Subsystem 5-436
Limit Line and Segment Subsystems 5-436
5-94 :SENSe{1-16}:ISEGMent{1-50} Subsystem 5-446
Limit Line and Segment Subsystems 5-446
5-95 :SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem. 5-452
Trigger, Hold, and External Source Subsystems. 5-452
General Procedure 5-452
5-96 :SENSe{1-16}:RECEiver Subsystem 5-485

5-97 :SENSe{1-16}:SEGMent Subsystem 5-487
Limit Line and Segment Subsystems 5-487
5-98 :SENSe{1-16}:SOURce{1-4} Subsystem 5-488
Trigger, Hold, and External Source Subsystems. 5-488
5-99 :SENSe{1-16}:SPUR Subsystem 5-489
Channel and Sweep Subsystems 5-489
5-100 :SENSe{1-16}:SWEep Subsystem 5-490
Channel and Sweep Subsystems 5-490
5-101 :SENSe{1-16}:TS3738 Subsystem. 5-494
5-102 :SENSe{1-16}:TS3739 Subsystem. 5-501
5-103 :SOURce:ALL:EXTernal Subsystem 5-508
Trigger, Hold, and External Source Subsystems. 5-508
5-104 :SOURce{1-16}:EFFective Subsystem. 5-509
Power Configuration Subsystems 5-509
5-105 :SOURce{1-16}:MODBB Subsystem 5-510
Power Configuration Subsystems 5-510
5-106 :SOURce{1-16}:POWer Subsystem 5-511
Power Configuration Subsystems 5-511
5-107 :SOURce{1-16}:TS3739 Subsystem. 5-523
Related Subsystems 5-523
5-108 :SOURce{1-4}:EXTernal Subsystem 5-526
Trigger, Hold, and External Source Subsystems. 5-526
5-109 :SOURce{1-16}:POWer Subsystem 5-528
5-110 :STATus:OPERation Subsystem 5-531
5-111 :STATus:QUEStionable Subsystem 5-533
5-112 :SYSTem Subsystem 5-536
5-113 :TRIGger[:SEQuence] Subsystem 5-552
Trigger, Hold, and External Source Subsystems. 5-552

Appendix A—Programming with LabVIEW

A-1 Introduction. A-1
A-2 Overview. A-1
Programming Basics A-1
GPIB Mnemonics. A-1
What is VISA? A-1
Programming Environments A-2
A-3 Installing the MS464X LabVIEW Driver A-2
Set up VectorStar Remote Interface A-5
A-4 A Quick Start with the Windows GPIB Utility A-6
References A-8
A-5 Programming Examples A-9
Getting Started A-9
A-6 Example 1 — Open a Session to get some Instrument Information A-16

A-7 Example 2 — Sending the *IDN? Command – Display Results A-19
A-8 Example 3 — Error Checking A-21
A-9 Example 4 — LIST Command – Send to a File A-22
A-10 Example 5 — Acquiring Trace Data A-25
A-11 Example 6 – Smith Chart Data A-27
A-12 Example 7 – Output S2P File A-30
A-13 Example 8 – Output BMP File A-31

Appendix B—Programming with LabWindows/CVI

B-1 Introduction B-1
B-2 Overview B-1
  Programming Basics B-1
  GPIB Mnemonics B-1
  What is VISA? B-1
  Programming Environments B-2
B-3 Installing the Anritsu au464x LabWindows/CVI Driver B-3
  Setup VectorStar for Native Language B-6
B-4 Quick Start with the Windows GPIB Utility B-7
  References B-8
B-5 Programming Examples B-9
  Getting Started B-9
  Setting up the Driver to Load every time CVI starts B-16
B-6 Example 1 – Open a Session to Obtain Instrument Information B-21
B-7 Example 2 – Sending the *IDN? Command and Displaying results B-24
B-8 Example 3 – Error Checking B-28
B-9 Example 4 — LIST Command – Send to a File B-33
B-10 Example 5 — Acquiring Trace Data B-38
B-11 Example 6 and 6a– Smith Chart Data B-43
  Example 6 – Acquire Smith Chart Trace Data B-43
  Example 6a – Smith Chart Output B-47
B-12 Example 7 – Output S2P File B-50
B-13 Example 8 – Output BMP File B-54

Appendix C—Programming with VISA/C#

C-1 Introduction C-1
C-2 Overview C-1
  Programming Basics C-1
  What is VISA? C-2
  Programming Environments C-4
  GPIB Mnemonics C-4
C-3 A Quick Start with the Windows GPIB Utility C-5
C-4 Programming Examples C-8
  Getting Started C-8

C-5 Example 1 — Open Session and *IDN? C-11
Example 1 – Code Listing C-11
Example 1 – Discussion C-13
C-6 Example 2 — Session Parameters and Status Checking C-16
Example 2 – Code Listing C-16
Example 2 – Discussion C-20
C-7 Example 3 — LIST Command C-22
Example 3 – Code Listing C-22
Example 3 – Discussion C-25
C-8 Example 4 — Acquiring ASCII Data, Arbitrary Block C-27
Example 4 – Code Listing C-27
Example 4 – Discussion C-31
C-9 Example 5 – Acquiring Binary Data, Arbitrary Block C-32
Example 5 – Code Listing C-32
Example 5 – Discussion C-36
C-10 Example 6 – Output S2P File C-37
Example 6 – Code Listing C-37
Example 6 – Discussion C-41
C-11 Example 7 – Output a Bitmap C-42
Example 7 – Code Listing C-42
Example 7 – Discussion C-46

Appendix D—Programming Basics with Legacy Software

D-1 Introduction D-1
D-2 Programming Basics D-1
VISA D-1
Programming Environments D-3
GPIB Mnemonics D-3
D-3 Installing the LabVIEW Driver D-4
D-4 Installing the LabWindows/CVI Driver D-7
D-5 Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser D-10

D-6 VISA and C# Programming Examples D-12
Create a New Console Application Project D-12
Build the Solution. D-13
Run the Code (Start Debugging). D-13
Example 1 – Opening a Session and Sending the *IDN? Command D-14
Example 1 Discussion D-15
Example 2 – Session Parameters and Status Checking D-17
Example 2 Discussion D-19
Example 3 – Sending Data to a File with the LIST Command. D-21
Example 3 Discussion D-23
Example 4 – Acquiring ASCII Data, Arbitrary Block D-25
Example 4 Discussion D-27
Example 5 – Acquiring Binary Data, Arbitrary Block D-28
Example 5 Discussion D-30
Example 6 – Output S2P File D-31
Example 6 Discussion D-33
Example 7 – Output a Bitmap D-34
Example 7 Discussion D-36

D-7 LabView Programming Examples. D-37
Setup Communications D-37
Example 1 – Opening a Session D-42
Example 2 – Sending the *IDN? Command D-44
Example 2 – Error Checking D-45
Example 3 – Sending Data to a File with the LIST Command. D-46
Example 4 – Acquiring Trace Data D-48
Example 5 – Smith Chart Data D-48
Example 6 – Output S2P File D-50
Example 7 – Output BMP File D-52

D-8 LabWindows/CVI Programming Examples. D-54
Example 0 – Opening a Session D-62
Example 1 – Sending the *IDN? Command and Displaying Results. D-63
Example 2 – Error Checking D-65
Example 3 – Sending Data to a File with the LIST Command. D-67
Example 4 – Acquiring Trace Data D-69
Example 5 – Smith Chart Data D-72
Example 6 – Output S2P File D-75
Example 7 – Output BMP File D-77

Appendix E—Alphabetical Command Index

E-1 Introduction E-1
Sorting E-1
System Suffix. E-1

E-2 Alphabetical Command Listing E-1

Chapter 1 — General Information

1-1 About this Manual

This manual provides information for remote operation of the VectorStar MS464xB Series VNAs using commands sent from an external controller via the IEEE 488 General Purpose Interface Bus (GPIB), USB, and Ethernet. It includes the following:

- A general description of the GPIB and the bus data transfer and control functions
- A listing of the IEEE 488 Interface Function Messages recognized by the VNA
- A brief description of the Ethernet and USB program interface to the VNA
- A complete listing and description of all the Standard Commands for Programmable Instruments (SCPI) commands that can be used to control VNA operation with examples of command usage
- See the companion document, Programming Manual Supplement – 10410-00323 for a complete listing of all Anritsu legacy Lightning (37xxxD/E) commands and all Agilent/HP legacy 8510 commands that can be used to control MS464xB Series VNAs.
- This manual is intended to be used in conjunction with the VectorStar MS464xB Series Microwave VNA Operation Manual – 10410-00317. Refer to that manual for general information about the VectorStar Series VNAs including equipment set up and front panel (manual mode) operating instructions.

Note	Many of the images in this document are used as typical representations of the product or of the product features. Your instrument and instrument displays may vary slightly from these images.

This manual covers the IEEE 488.2, System and Troubleshooting, and SCPI commands. See the Programming Manual - Supplement, 10410-00323 for descriptions of the Anritsu 37xxxx and HP8510 commands.

1-2 Introduction

This chapter provides a general description of the GPIB and the bus data transfer and control functions. It also contains a listing of the MS464xB GPIB interface function subset capability and response to IEEE 488 interface function messages.

The GPIB information presented in this chapter is general in nature. For complete and specific information, refer to the following documents, available from the *Institute of Electrical and Electronics Engineers*:

- ANSI/IEEE Standard 488.1-1987 IEEE Standard Digital Interface for Programmable Instrumentation
- ANSI/IEEE Standard 488.2-1987 IEEE Standard Codes, Formats, Protocols, and Common Commands

These documents precisely define the total specification of the mechanical and electrical interface, and of the data transfer and control protocols.

The final section in this chapter, “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22, provides a listing of the VNA instrument minimum and maximum frequency settings and related parameters such as default frequency span. These values are provided for standalone VectorStar VNAs as well as for VNAs configured with ME7828A or MS7838A Broadband/Millimeter-Wave/Multiple Source Test Sets.

Note	When operating the VectorStar VNA through remote programming, the front panel user interface and controls are disabled. To return to local front panel control, press the front panel Clear/Tab key [**Clr -->|**], keyboard **Esc** key, or send the `RTL` command. For general information about GPIB, refer to Section 1-5 “IEEE 488 GPIB Description”.

1-3 Related Documentation

The latest product information and documentation can be found on the VectorStar product web page:

http://www.anritsu.com/en-us/products-solutions/products/ms4640b-series.aspx

On this web page, you can select various tabs for more information about your instrument. Included is a “Library” tab which contains links to all the latest technical documentation related to this instrument.

Product Information, Compliance, and Safety

Refer to the VectorStar Product Information, Compliance, and Safety (PICS) – 10100-00063 for applicable product information, compliance statements, and safety information, including links to applicable product web pages.

1-4 Contacting Anritsu

To contact Anritsu, please visit:

http://www.anritsu.com/contact.asp

From here, you can select the latest sales, service and support contact information in your country or region, provide online feedback, complete a "Talk to Anritsu" form to get your questions answered, or obtain other services offered by Anritsu.

1-5 IEEE 488 GPIB Description

The IEEE 488 General Purpose Interface Bus (GPIB) is an international instrumentation interface standard for integrating instruments, computers, printers, plotters, and other measurement devices into systems. IEEE stands for the Institute of Electrical and Electronics Engineers and is currently the world's largest technical professional society. Refer to http://www.ieee.org for more information about the society and its standards.

The interface between the VectorStar VNA and other devices on the GPIB is via a standard 24-wire GPIB interface cable. This cable uses a double-sided connector where one connector face is a plug and the other a receptacle. These double-function connectors allow parallel connection of two or more cables to a single instrument connector.

Caution	When two or more cables are connected in parallel, be careful not to bend the attached connectors and damage the rear panel GPIB Port connector.

GPIB Network Restrictions

To achieve design performance on the GPIB network bus, the proper timing and voltage level relationships must be maintained. If either the cable length between separate instruments or the accumulated cable length between all instruments is too long, the data and control lines cannot be driven properly and the system may fail to perform.

- **Network Topology**

  GPIB network topologies can be in any combination of star and/or serial ("daisy-chain") network topology subject to the limitations below. Looped topologies are prohibited. If a star topology is used, care must be observed when stacking two or more GPIB connectors.

- **Number of Units**

  The maximum number of physical GPIB devices must be less than or equal to 15 including the controller.

- **Addresses**

  The default address for a GPIB Controller Device must be 0 (zero). The address for a GPIB-controlled device can be from 1 (one) to 30. Each device address in a GPIB network must be unique.

- **Total Network Length**

  The network length is measured in meters. The total network length of all cables less than two times the number of instruments, and always less than 20 meters. For example, the maximum length of a 2 instrument network is 4 meters; a 6 instrument network is limited to 12 meters; and networks of 10 or more instruments are limited to 20 meters.

- **Maximum Network Leg Length**

  The recommended length of any network leg is less than or equal to 4 meters. In all cases, minimize cable lengths to achieve maximum data transfer rates.

- **Power State**

  Two-thirds of the network devices must be powered on before signaling is started. No devices should be powered up while bus signaling is in operation where devices are actively sending or receiving messages or data.

Note	For low EMI applications, the GPIB cable should be a fully shielded type with well-grounded metal-shell connectors. Use Anritsu Model 2100-series cables.

The devices on the GPIB are connected in parallel, as shown in Figure 1-1. The interface consists of 16 signal lines and 8 ground lines in a shielded cable. Eight of the signal lines are the data lines, DIO 1 through DIO 8. These data lines carry messages (data and commands), one byte at a time, among the GPIB devices. Three of the remaining lines are the handshake lines that control the transfer of message bytes between devices. The five remaining signal lines are referred to as interface management lines.

The following paragraphs provide an overview of the GPIB including descriptions of:

- "Functional Elements"
- "Bus Structure"
- "Data Bus Description"
- "Data Byte Transfer Control Bus Description"
- "General Interface Management Bus Description"
- "Device Interface Function Capability"
- "Message Types"
- "Response to GPIB Interface Function Messages"
- "Configuring the Dedicated GPIB Port"
- "Configuring the IEEE 488.2 GPIB Port"

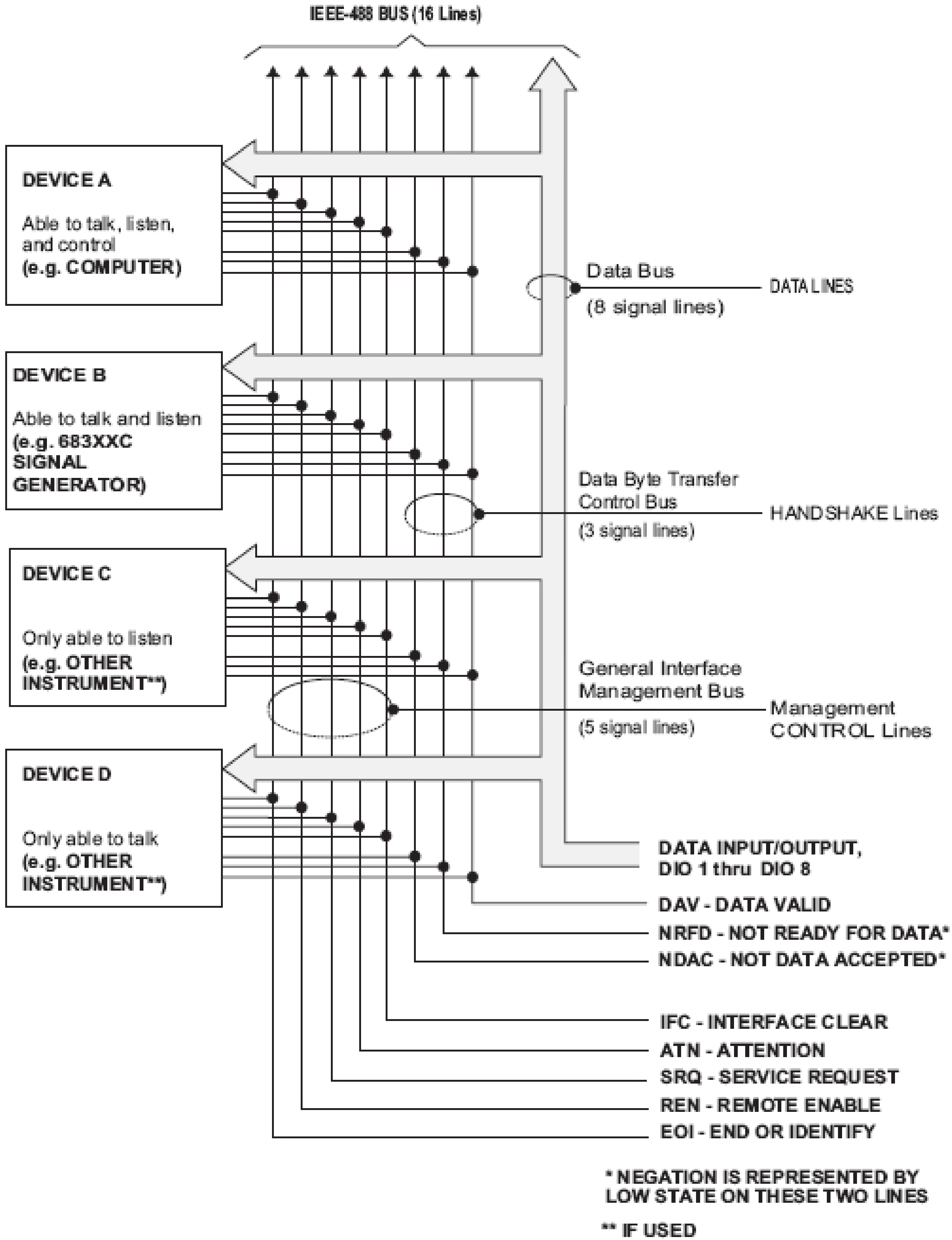


Figure 1-1. GPIB Interface Connection and Bus Structure

Functional Elements

Effective communications between devices on the GPIB requires three functional elements; a talker, a listener, and a controller. Each device on the GPIB is categorized as one of these elements depending on its current interface function and capabilities.

- **Talker**

  A talker is a device capable of sending device-dependent data to another device on the bus when addressed to talk. Only one GPIB device at a time can be an active talker.

- **Listener**

  A listener is a device capable of receiving device-dependent data from another device on the bus when addressed to listen. Any number of GPIB devices can be listeners simultaneously.

- **Controller**

  A controller is a device, usually a computer, capable of managing the operation of the GPIB. Only one GPIB device at a time can be an active controller. The active controller manages the transfer of device-dependent data between GPIB devices by designating who will talk and who will listen.

- **System Controller**

  The system controller is the device that always retains ultimate control of the GPIB. When the system is first powered-up, the system controller is the active controller and manages the GPIB. The system controller can pass control to a device, making it the new active controller. The new active controller, in turn, may pass control on to yet another device. Even if it is not the active controller, the system controller maintains control of the Interface Clear (IFC) and Remote Enable (REN) interface management lines and can thus take control of the GPIB at anytime.

Bus Structure

The GPIB uses 16 signal lines to carry data and commands between the devices connected to the bus. The interface signal lines are organized into three functional groups.

- Data Bus (8 lines)
- Data Byte Transfer Control Bus (3 lines)
- General Interface Management Bus (5 lines)

The signal lines in each of the three groups are designated according to function. Table 1-1 lists these designations.

Table 1-1. Interface Bus Signal Line Designations

Bus Type	Signal Line Name	Function
Data Bus	DIO1–DIO8	Data Input/Output, 1 through 8
Data Byte Transfer Control Bus	DAV NRFD NDAC	Data Available Not Ready For Data Not Data Accepted
General Interface Management Bus	ATN IFC SRQ REN EOI	Attention Interface Clear Service Request Remote Enable End Or Identify

Data Bus Description

The data bus is the conduit for the transfer of data and commands between the devices on the GPIB. It contains eight bidirectional, active-low signal lines —DIO 1 through DIO 8. Data and commands are transferred over the data bus in byte-serial, bit-parallel form. This means that one byte of data (eight bits) is transferred over the bus at a time. DIO 1 represents the least-significant bit (LSB) in this byte and DIO 8 represents the most-significant bit (MSB). Bytes of data are normally formatted in seven-bit ASCII (American Standard Code for Information Interchange) code. The eighth (parity) bit is not used.

Each byte placed on the data bus represents either a command or a data byte. If the Attention (ATN) interface management line is TRUE while the data is transferred, then the data bus is carrying a bus command which is to be received by every GPIB device. If ATN is FALSE, then a data byte is being transferred and only the active listeners will receive that byte.

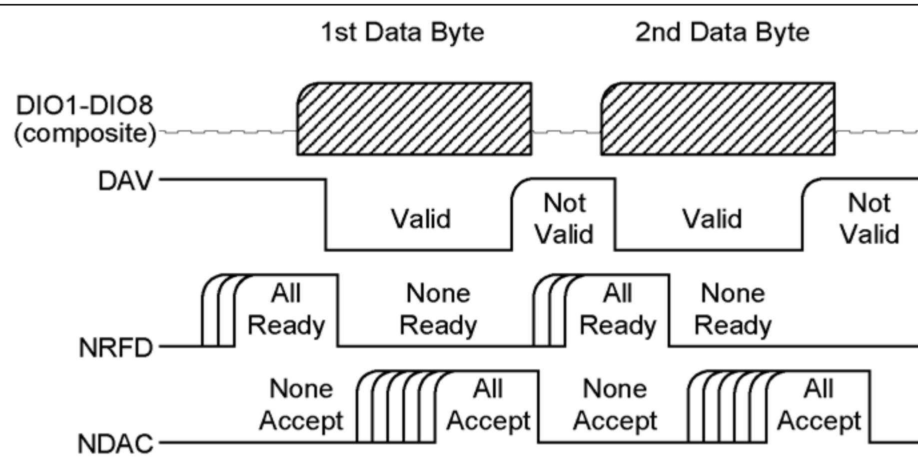


Figure 1-2. Typical GPIB Handshake Operation

Data Byte Transfer Control Bus Description

Control of the transfer of each byte of data on the data bus is accomplished by a technique called the "three-wire handshake", which involves the three signal lines of the Data Byte Transfer Control Bus. This technique forces data transfers at the speed of the slowest listener, which ensures data integrity in multiple listener transfers. One line (DAV) is controlled by the talker, while the other two (NRFD and NDAC) are wired-OR lines shared by all active listeners. The handshake lines, like the other GPIB lines, are active low. The technique is described briefly in the following paragraphs and is depicted in Figure 1-2. For further information, refer to the ANSI/IEEE Standard 488.1 specification.

General Interface Management Bus Description

The general interface management bus is a group of five signal lines used to manage the flow of information across the GPIB. A description of the function of each of the individual control lines is provided below.

- **ATN (Attention)**

  The active controller uses the ATN line to define whether the information on the data bus is a command or is data. When ATN is TRUE (low), the bus is in the command mode and the data lines carry bus commands. When ATN is FALSE (high), the bus is in the data mode and the data lines carry device-dependent instructions or data.

- **EOI (End or Identify)**

  The EOI line is used to indicate the last byte of a multi byte data transfer. The talker sets the EOI line TRUE during the last data byte. The active controller also uses the EOI line in conjunction with the ATN line to initiate a parallel poll sequence.

- **IFC (Interface Clear)**

  Only the system controller uses this line. When IFC is TRUE (low), all devices on the bus are placed in a known, quiescent state (unaddressed to talk, unaddressed to listen, and service request idle).

- **REN (Remote Enable)**

  Only the system controller uses this line. When REN is set TRUE (low), the bus is in the remote mode and devices are addressed either to listen or to talk. When the bus is in remote and a device is addressed, it receives instructions from the GPIB rather than from its front panel. When REN is set FALSE (high), the bus and all devices return to local operation.

- **SRQ (Service Request)**

  The SRQ line is set TRUE (low) by any device requesting service by the active controller.

Device Interface Function Capability

An interface function is the GPIB system element which provides the basic operational facility through which a device can receive, process, and send messages. Each specific interface function may only send or receive a limited set of messages within particular classes of messages. As a result, a set of interface functions is necessary to achieve complete communications among devices on the GPIB.

The ANSI/IEEE Standard 488.1 specification defines each of the interface functions along with its specific protocol. The ANSI/IEEE Standard 488.2 specification specifies the minimum set of IEEE 488.1 interface capabilities that each GPIB device must have. This minimum set of interface functions assures that the device is able to send and receive data, request service, and respond to a device clear message. Table 1-2 lists the interface function capability of the VectorStar Series VNA.

Table 1-2. VectorStar VNA Interface Function Capability

Function Identifier	Function	VectorStar VNA Capability
AH1	Acceptor Handshake	Complete Capability
SH1	Source Handshake	Complete Capability
T6	Talker	No Talk Only (TON)
L4	Listener	No Listen Only (LON)
SR1	Service Request	Complete Capability
RL1	Remote/Local	Complete Capability
PP1	Parallel Poll	Complete Capability
DC1	Device Clear	Complete Capability
DT1	Device Trigger	Complete Capability
C0 C1 C2 C3 C28	Controller Capability Options	C0, No Capability C1, System Controller C2, Send IFC and Take Charge C3, Send REN C28, Send IF Messages
E2	Tri-State Drivers	Three-state bus drivers

Message Types

There are three types of information transmitted over the GPIB:

- interface function messages
- device-specific commands, and
- data and instrument status messages.

Interface Function Messages

The controller manages the flow of information on the GPIB using interface function messages, usually called commands or command messages. Interface function messages perform such functions as initializing the bus, addressing and unaddressing devices, and setting device modes for remote or local operation.

There are two types of commands: multiline and uniline. Multiline commands are bytes sent by the active controller over the data bus (DIO1-DIO8) with ATN set TRUE. Uniline commands are signals carried by the individual interface management lines.

The user generally has control over these commands; however, the extent of user control depends on the implementation and varies with the specific GPIB interface hardware and software used with the external controller.

Device-Specific Commands

These commands are keywords or mnemonic codes sent by the external controller to control the setup and operation of the addressed device or instrument. The commands are normally unique to a particular instrument or class of instruments and are described in its documentation. Device-specific commands are transmitted over the data bus of the GPIB to the device in the form of ASCII strings containing one or more keywords or codes. They are decoded by the device's internal controller and cause the various instrument functions to be performed.

Data and Instrument Status Messages

These messages are sent by the device to the external controller via the GPIB. They contain measurement results, instrument status, or data files that the device transmits over the data bus in response to specific requests from the external controller. The contents of these messages are instrument specific and may be in the form of ASCII strings or binary data. In some cases data messages will be transmitted from the external controller to the device. For example, messages to load calibration data.

An SRQ (service request) is an interface function message sent from the device to the external controller to request service from the controller, usually due to some predetermined status condition or error. To send this message, the device sets the SRQ line of the General Interface Management Bus true, then sends a status byte on the data bus lines.

An SRQ interface function message is also sent by the device in response to a serial poll message from the controller, or upon receiving an Output Status Byte(s) command from the controller. The protocols associated with the SRQ functions are defined in the ANSI/IEEE Std 488.2 document.

The manner in which interface function messages and device-specific commands are invoked in programs is implementation specific for the GPIB interface used with the external controller. Even though both message types are represented by mnemonics, they are implemented and used in different ways.

Normally, the interface function messages are sent automatically by the GPIB driver software in response to invocation of a software function. For example, to send the IFC (Interface Clear) interface function message, one would call the ibsic function of the National Instruments software driver. On the other hand, the command *RST (Reset) is sent in a command string to the addressed device. In the case of the National Instruments example, this would be done by using the ibwrt function call.

Response to GPIB Interface Function Messages

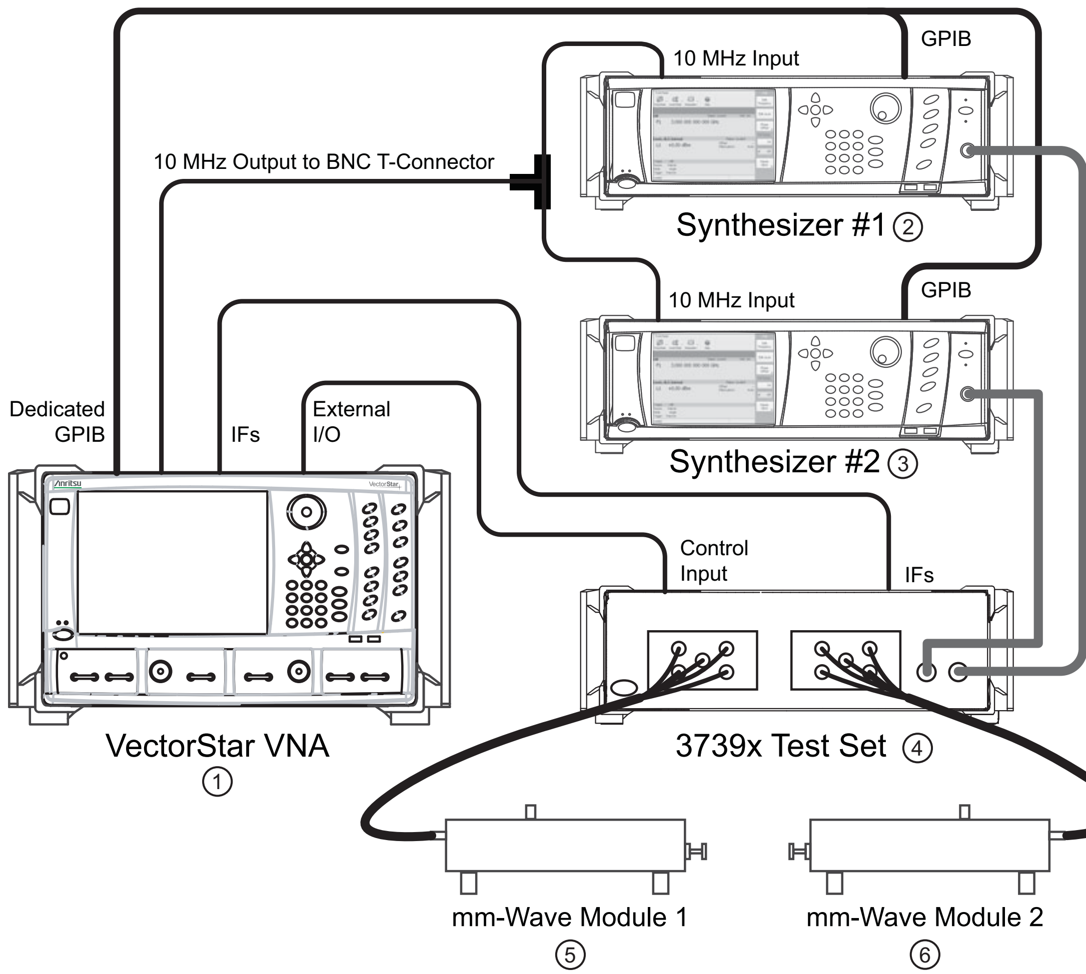
Table 1-3 lists the GPIB interface function messages that the MS464xB will recognize and respond to. With the exception of the Device Clear and Selected Device Clear messages, these messages affect only the operation of the VectorStar VNA GPIB interface. The VNA response for each message is indicated below. Interface function messages are transmitted on the GPIB data lines and interface management lines as either unaddressed or addressed to receive the commands. The manner in which these messages are invoked in programs is implementation dependent. For more programming information, refer to the documentation included with the GPIB interface used for the external controller and to the IEEE 488.1 specification.

Table 1-3. VectorStar VNA Response to GPIB Interface Function Messages

Interface Function Message	Addressed Command	VectorStar VNA Response
Device Clear (DCL) Selected Device Clear (SDC)	No Yes	Clears the input and output buffers and resets the parser.
Go To Local (GTL)	Yes	Returns the VNA to local front panel control.
Group Execute Trigger (GET)	Yes	In step sweep, single trigger: triggers a sweep. In list sweep, manual trigger: triggers step to next index within start/stop index boundaries. In list sweep, single trigger: triggers a sweep.
Interface Clear (IFC)	No	Stops the VNA GPIB interface from listening or talking. (The front panel controls are not cleared.)
Local Lockout (LLO)	No	Disables the front panel menu RETURN TO LOCAL soft-key.
Remote Enable (REN)	No	Places the VNA under remote (GPIB) control when it has been addressed to listen.
Serial-Poll Enable (SPE)	No	Outputs the serial-poll status byte.
Serial-Poll Disable (SPD)	No	Disables the serial-poll function.
Parallel-Poll Configure (PPC)	Yes	The VectorStar VNA does not respond to Parallel-Poll messages.
Parallel-Poll Unconfigure	No	The VectorStar VNA does not respond to Parallel-Poll messages.

Configuring the Dedicated GPIB Port

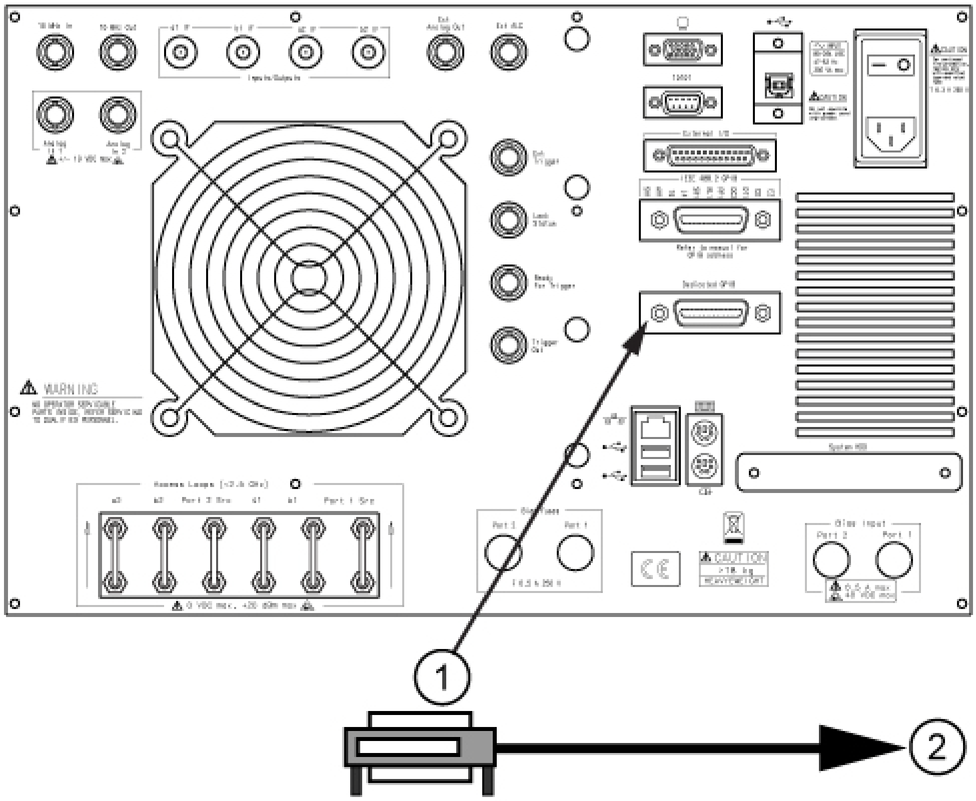
Use this procedure to setup the dedicated GPIB port to control other GPIB devices such as Power Meters or Signal Generators by using the VNA rear panel **Dedicated GPIB Port**. An example of a complex GPIB configuration with the VNA as the GPIB controller is shown in Figure 1-3.



1. VectorStar VNA	4. 3739x Broadband Test Set
2. Synthesizer #1	5. Millimeter-Wave (mm-Wave) Module #1
3. Synthesizer #2	6. mm-Wave Module #2

Figure 1-3. VNA as a Remote GPIB Controller for Synthesizers

1. Connect GPIB devices to the VNA rear panel **Dedicated GPIB Port** using the appropriate length GPIB cable.



1. **GPIB Port:** Rear panel Dedicated GPIB Port D-24 (f) connector.	2. **GPIB Cable and Connector:** GPIB cable and double-sided connector. Cable routes to any controlled GPIB devices.

Figure 1-4. VNA Dedicated GPIB Port and GPIB Cable Connection

2. Power-up the VectorStar VNA and any attached GPIB devices, and allow the system to warm up.
3. Navigate to the Remote Inter. menu as follows:

   MAIN | System > | SYSTEM | Remote Interface > | REMOTE INTER/

.

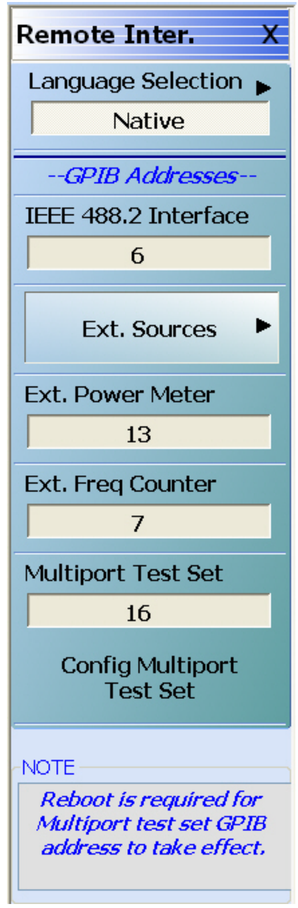


Figure 1-5. REMOTE INTERFACE (REMOTE INTER) Menu

The GPIB address for any controlled device can be kept at the factory default values or changed as required. The default GPIB addresses are:

- VectorStar MS464xB Series VNA = 6
- VectorStar MN469xC Series Test Set = 16
- External Source 1 = 4
- External Source 2 = 5
- External Source 3 = 2
- External Source 4 = 3
- Power Meter = 13
- Frequency Counter = 7
- W-Band Power Meter = 15

The GPIB addresses for the External Sources can be changed from the Ext Src Addr sub-menu available at:

- MAIN | System | SYSTEM | Remote Interface | REMOTE INTER | Ext Sources | EXT SRC ADDR.

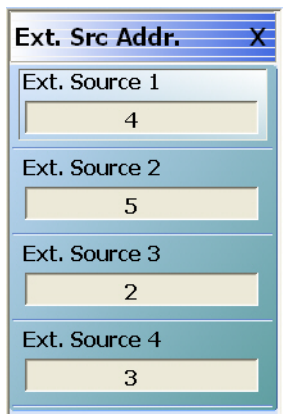


Figure 1-6. EXT SRC ADDR (EXTERNAL SOURCE ADDRESS) Menu

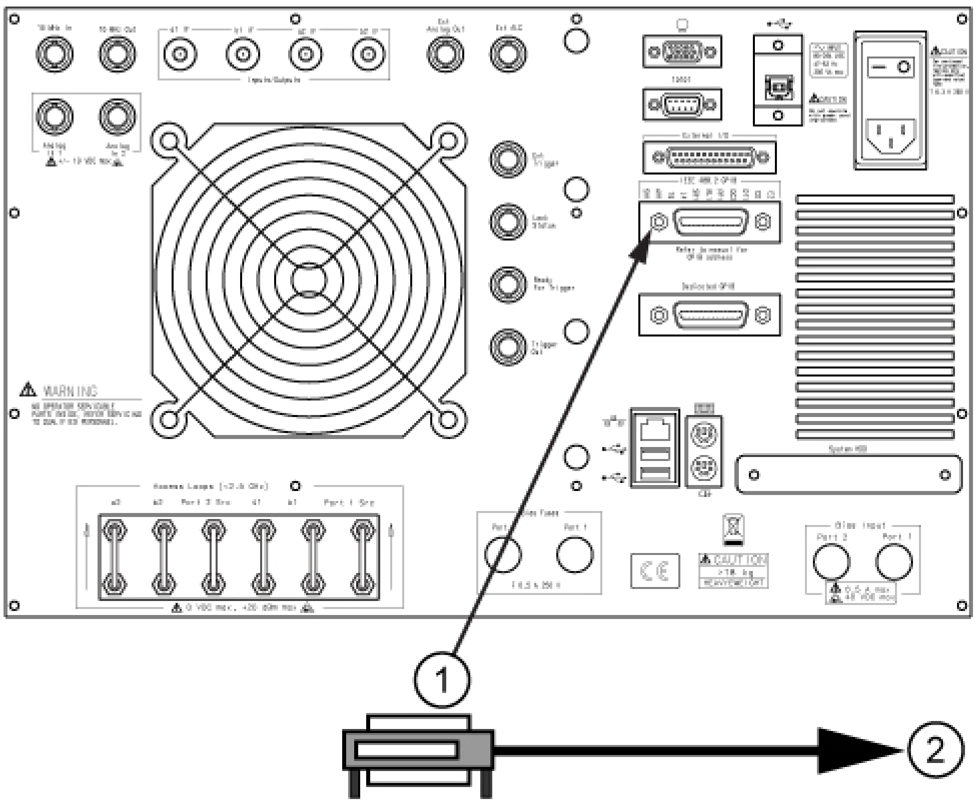
4. If a change of GPIB address is required, click the device button, and the device field toolbar appears as shown below:

5. Use the front panel keys, a keyboard, or mouse to set the required GPIB address.
6. Click the Enter button to set the new GPIB address.

Configuring the IEEE 488.2 GPIB Port

Use this procedure to set up the talker/listener port to control the VNA remotely over an IEEE 488.2 GPIB bus network. In this configuration, an external computer/controller is physically connected to the VNA rear panel **IEEE 488.2 GPIB Port** with a standard GPIB cable. Most of the VectorStar VNA functions (except power on/off and initialization of the hard disk) can be controlled remotely by an external computer/controller via the IEEE-488.2 GPIB bus network.

1. Connect the VNA to the GPIB Controller using the rear panel **IEEE 488.2 GPIB Port** DB-24 (f) connector and a standard GPIB cable.



1. **IEEE 488.2 Talker/Listener Port:** Rear panel IEEE 488.2 Talker/Listener GPIB Port D-24 (f) connector.	2. **GPIB Cable and Connector:** GPIB cable and double-sided connector. Cable routes to the GPIB Controller computer.

Figure 1-7. IEEE 488.2 GPIB Talker/Listener Port and GPIB Cable Connection

2. Apply power to the VectorStar VNA and allow the system to warm up.

Once the software has finished loading and start-up testing is complete, the instrument is ready to be remotely controlled via the GPIB. The instrument will not respond to GPIB commands until the instrument's software has been loaded.

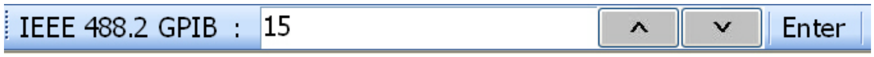
Note	The factory default GPIB address for the instrument is six (6).

To change the VectorStar VNA GPIB address:

1. Navigate to the REMOTE INTER menu:

   MAIN | System | SYSTEM | Remote Interface | REMOTE INTER
2. On the Remote Inter. menu, the IEEE 488.2 GPIB button.

   The IEEE 488.2 GPIB toolbar appears below the icon toolbar near the top of the display.



3. Use the front panel keys, a keyboard, or a mouse to set the required GPIB address.
4. Click the Enter button to set the new GPIB address.

1-6 Resetting GPIB Addresses

The VectorStar VNA Reset commands received over the GPIB do not reset GPIB addresses. Below is a list of the Reset commands which are recognized by the VectorStar VNA.

- IEEE488 Commands
  - *RST - Instrument reset
- Lightning Commands
  - RST - Instrument reset
  - RST0 - Instrument reset
  - RST1 - Instrument reset
- HP8510 Commands
  - FACTPRES - Perform a factory preset
  - PRES - Instrument reset
- Native SCPI Commands
  - :SYSTem:PRESet - Performs Preset
  - :SYSTem:PRESet:ZERo - Performs a Reset Zero

Background

ATE programs use a Reset command to return the VNA to a known state. Most large ATE programs which do multiple operations use a Reset command at the start of each operation. However, allowing a Reset command to alter GPIB addresses could unexpectedly cause loss of communication with the External Controller and/or the External Instruments on the VNA Dedicated Bus. Therefore the reset commands listed above DO NOT alter GPIB addresses.

Returning the GPIB Addresses to Default

If it is necessary to return the GPIB addresses associated with the VectorStar VNA to their normal default values, use the following procedure.

1. Exit the VectorStar VNA application by selecting MENU BAR | File | Exit at the top of the Display. This will return execution to the PC desktop.
2. Start the VectorStar Reset Application by double clicking its Icon (Yellow Starburst) on the Desktop.
3. When the VectorStar Reset Application starts up, click the Reset VectorStar button. This will sequentially reset the VectorStar application and the VectorStar 100K application.
4. When done, exit the application.
5. VectorStar VNA operation can be resumed by double clicking the Blue VectorStar Icon on the Desktop.

1-7 Ethernet LAN TCP/IP and USB Description

The VectorStar VNA supports Ethernet 10/100 BASE-T. The instrument is connected directly to the LAN via the rear panel RJ-45 Ethernet Port using a standard CAT-5 Ethernet cable.

The MS464xB Series VNAs can also be controlled remotely across a Universal Serial Bus (USB) Type 2 network by a PC equipped with a USB control application using the rear panel **USB Control Port**.

The general requirements for manual Ethernet LAN configuration are discussed in the sections below.

Note	This section is provided for general information about manually configuring an Ethernet connection and does not apply to USB networking. Consult your local network administrator for the exact requirements and settings that are required for your network installation.

TCP/IP General Requirements and Settings

Transmission Control Protocol/Internet Protocol (TCP/IP) is a network protocol. In the Windows operating system, TCP/IP is automatically installed and in most cases, installation, configuration, and communication are transparent to the user.

In a TCP/IP network, you must provide IP addresses and other information to clients. Clients may also require a naming service or a method for name resolution. The TCP/IP protocol setup requires the following information:

- **IP Address**

  Every device in a TCP/IP network requires an IP address that consists of four numbers, each between 0 and 255, separated by periods. For example: 128.111.122.42 is a valid IP address.

- **Subnet Mask**

  The subnet mask distinguishes the portion of the IP address that is the network identification (ID) address from the portion that is the station ID address. When the subnet mask 255.255.0.0 is applied to the IP address above, it would identify the network ID address as 128.111 and the station ID address as 122.42. All stations in the same Local Area Network (LAN) should have the same network ID, but different station IDs.

- **Default Gateway**

  A TCP/IP network can have a gateway to communicate beyond the LAN identified by the network ID. A gateway is a computer or electronic device that is connected to two different networks and can move TCP/IP data from one network to the other. A single LAN that is not connected to other LANs requires a default gateway setting of 0.0.0.0. The default gateway setting for the VectorStar MS4640B Series VNA is 0.0.0.0. If your network has a gateway, then the default gateway would be set to the appropriate value of your gateway.

- **Hardware Address (MAC Address)**

  An Ethernet address is a unique 48-bit value that identifies a network interface card internal to the VNA to the rest of the network. Every network card has a unique Ethernet address permanently stored into its memory.

- **TCP/IP Port Number**

  Reports the currently used TCP/IP port number with Port #5001 set as the default. In general, it should not be changed. If the port number is changed, do not change it to 5000 as that port is used by VXI-11. Ports below #5000 are generally reserved for other services and devices. Custom settings generally should use settings of #5001 and higher.

- **Network Interface Setup**

  TCP/IP connectivity requires setting up the parameters described at the beginning of this section. You may need to contact your network administrator or refer to your network documentation for further assistance. The following procedure is a general overview of how to set up a general LAN connection on both the VNA and the remote machine. The actual menus and sequence may vary.

Default Plug-and-Play Configuration

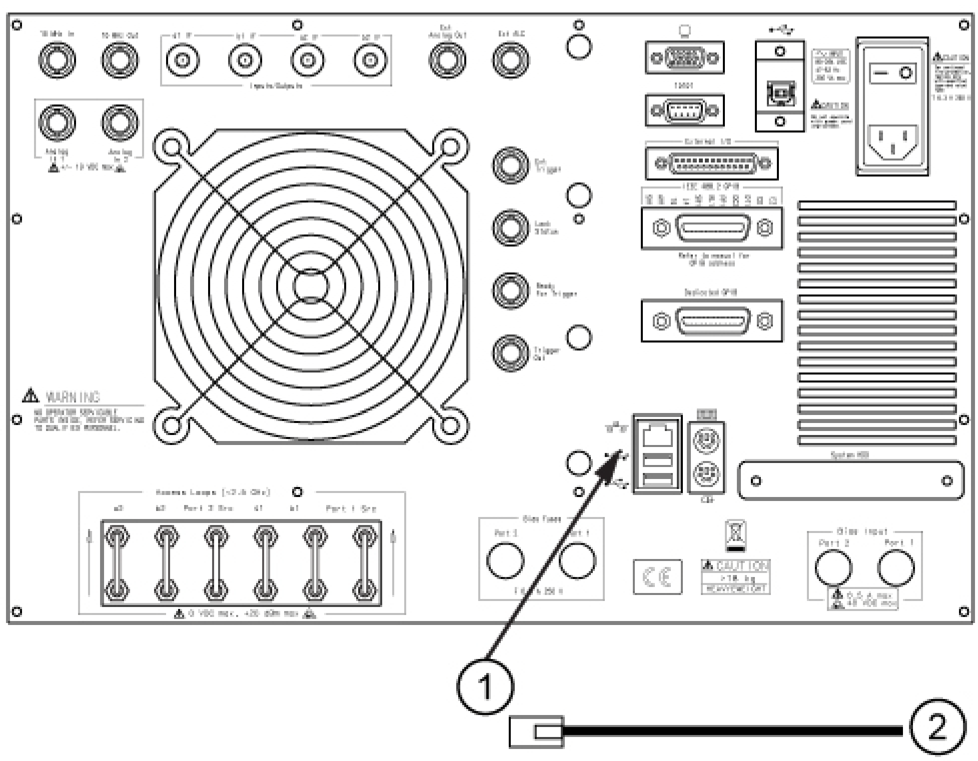
The VectorStar VNA and its Windows operating system comes pre-configured and ready to plug into and connect with an existing Ethernet network. The Ethernet network must provide DNS/DHCP and be configured with a gateway.

Connection can also be made to a USB network. USB networks are typically auto-detecting without the need for configuration; however, a special USB network cable (described below) is required.

The required PC-to-Instrument USB 2.0 networking cable (also called a USB-USB cable, USB networking cable, or USB bridge cable) has a small electronic circuit module in the middle of the cable that allows the PC and the VectorStar VNA to talk to each other at 480 Mbps. The PC-to-Instrument USB cable is user-provided and does not come with the VectorStar VNA.

Warning	Using the wrong type of USB-USB cable can result in damage to the PC and/or VectorStar USB ports and/or power supplies. Do NOT use a so-called “USB A/A” cable that is similar to a serial port cross-over null-modem cable.

For Ethernet connection, connect the VectorStar VNA to the network with an Ethernet cable between the VNA rear panel RJ-45 Ethernet Port and your local network port. For USB networks, use the USB network cable defined above between the VNA rear panel USB Type 2 Port and your local network hub.



1. Ethernet Port – Rear panel Ethernet RJ45 (f) port or USB Type 2 connection	2. Ethernet CAT-5 RJ-45 or USB cable from LAN (Local Area Network).

Figure 1-8. VectorStar VNA Network Connection

Windows will automatically detect the network settings and configure the network connection.

Manually Configuring TCP/IP Ethernet LAN Settings

To see the current network settings for your VectorStar VNA, with the VectorStar Application Software running, navigate to the Network Interface (Network Interf.) menu as follows:

- MAIN | System | SYSTEM | Ntwk Interface | NETWORK INTERF.

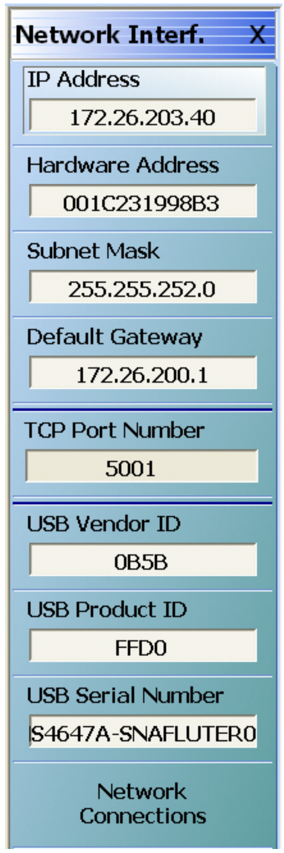


Figure 1-9. NETWORK INTERFACE (NETWORK INTERF) Menu

The top eight (8) display buttons provide information for the current network settings. Changes to these settings must be made through the Microsoft Windows configuration utilities by clicking Network Connections.

The Network Connections dialog box shows the current available local networks.

Note	You may need to consult your network documentation or network administrator for assistance in manually configuring your network setup. The Windows network connections Help may provide information related to computer networking.

1-8 Configuring the Remote Language

The VectorStar VNA command parser responds to Native SCPI commands, legacy Anritsu Lightning commands, or to HP8510 commands.

When the programming language is set to Native:

- The VectorStar VNA responds to Lightning commands on a secondary basis.
- Does not respond to HP8510 commands.

When the programming language is set to Lightning:

- The VectorStar VNA responds to SCPI commands on a secondary basis.
- Does not respond to HP8510 commands.

When the language is set to HP8510:

- The VectorStar VNA responds to SCPI commands on a secondary basis.
- And responds to Lightning commands on a tertiary basis.

Processing and response times are improved if the Remote Language setting matches the majority of the sent commands. Scripts written in older versions of the Lightning command set or in the HP/Agilent 8510 command set may need editing to be fully compatible with the VNA. Not all Lightning and HP/Agilent commands are supported.

Procedure

If necessary, you can change the VNA default GPIB language from Native to Lightning or HP8510 as follows:

1. Navigate to the Remote Lang. menu:

   MAIN | System | SYSTEM | Remote Interface | REMOTE INTER | Language Selection | REMOTE LANG

Figure 1-10. REMOTE LANGUAGE (REMOTE LANG) Menu

2. Select the interface language being used.
3. Click Back at the bottom of the menu to return to the REMOTE INTER menu.

1-9 Minimum/Maximum Instrument Frequency and Related Parameters

The minimum and maximum instrument frequencies depend on the instrument model and the installed options. The general frequency limits for the :SENSe{1-16}:FREQuency subsystem and related commands are defined below in Table 1-4, Table 1-5, Table 1-6, Table 1-7, and Table 1-8.

In general, the frequency default values and limits are affected by three factors:

- Instrument model number, which defines the VNA standalone instrument high-side frequency limit
- Installed options (Option 70), if equipped, which define the VNA instrument low-side limit. On MS464xB Series VNAs, Option 70 provides a lower limit specified to 70 kHz, but which is allowed to go to 40 kHz.
- Attached broadband, millimeter-wave, and/or multiple source equipment (if equipped) that defines the VNA system high-side frequency limit.

Standalone VNAs

For the base VNA, the frequency range is established by the models (for the high end) and options (for the low end). For the higher frequency systems, the limits are dependent on the exact hardware being used and the instrument mode.

The tables below provide standalone VNA frequency limits:

- Table 1-4, "Standalone VNAs – Default Start, Default CW, and Default Stop Frequencies" on page 1-23
- Table 1-5, "Standalone VNAs – Minimum Start, Minimum CW, and Maximum Start Frequencies" on page 1-24
- Table 1-6, "Standalone VNAs – Minimum Stop, Maximum Stop, and Maximum CW Frequencies" on page 1-25
- Table 1-7, "Standalone VNAs – Default Frequency Span and Maximum Frequency Span" on page 1-26
- Table 1-8, "Standalone VNAs – Minimum Center Frequency and Maximum Center Frequency" on page 1-27
- Table 1-9, "Standalone VNAs – Default Center Frequencies" on page 1-28

Standalone VNAs – Default Start, Default CW, and Default Stop Frequencies

The instrument start frequency depends if the Option 70 – 70 kHz Low End Frequency Extension is installed.

Table 1-4. Standalone VNAs – Default Start, Default CW, and Default Stop Frequencies

Model	Default Start Frequency or Default CW Frequency without Option 070	Default Start Frequency or Default CW Frequency with Option 70	Default Stop Frequency
MS4642B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	2.0000000000E+010 20 GHz
MS4644B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	4.0000000000E+010 40 GHz
MS4645B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	5.0000000000E+010 50 GHz
MS4647B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	7.0000000000E+010 70 GHz

Standalone VNAs – Minimum Start, Minimum CW, and Maximum Start Frequencies

The highest possible setting for the Start Frequency is the Stop Frequency minus 2 Hz. This yields a sweep with three data points.

Table 1-5. Standalone VNAs – Minimum Start, Minimum CW, and Maximum Start Frequencies

Model	Minimum Start Frequency or Minimum CW Frequency without Option 070	Minimum Start Frequency or Minimum CW Frequency with Option 070	Maximum Start Frequency (Start Frequency – 2 Hz)
MS4642B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	1.9999999998E+010 20 GHz – 2 Hz
MS4644B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	3.9999999998E+010 40 GHz – 2 Hz
MS4645B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	4.9999999998E+010 50 GHz – 2 Hz
MS4647B	1.0000000000E+007 10 MHz	7.0000000000E+004 70 kHz	6.9999999998E+010 70 GHz – 2 Hz

Standalone VNAs – Minimum Stop and Maximum Stop Frequencies

The lowest possible setting for the stop frequency is the start frequency plus 2 Hz which yields as sweep of three (3) data points.

Table 1-6. Standalone VNAs – Minimum Stop, Maximum Stop, and Maximum CW Frequencies

Model	Minimum Stop Frequency without Option 70 Stop Min = Start + 2 Hz = 3 data points	Minimum Stop Frequency with Option 070 Stop Min = Start + 2 Hz = 3 data points	Maximum Stop Frequency or Maximum CW Frequency Stop Max = Instrument Max)
MS4642B	1.0000002000E+007 10 MHz + 2 Hz	7.0002000000E+004 70 kHz + 2 Hz	2.0000000000E+010 20 GHz
MS4644B	1.0000002000E+007 10 MHz + 2 Hz	7.0002000000E+004 70 kHz + 2 Hz	4.0000000000E+010 40 GHz
MS4645B	1.0000002000E+007 10 MHz + 2 Hz	7.0002000000E+004 70 kHz + 2 Hz	5.0000000000E+010 50 GHz
MS4647B	1.0000002000E+007 10 MHz + 2 Hz	7.0002000000E+004 70 kHz + 2 Hz	7.0000000000E+010 70 GHz

Standalone VNAs – Default Frequency Span and Maximum Frequency Span

The frequency span equals the stop frequency minus the start frequency. The minimum possible frequency span is frequency is 2 Hz.

Table 1-7. Standalone VNAs – Default Frequency Span and Maximum Frequency Span

Model	Minimum Frequency Span All Models Span Min = 2 Hz	Default Frequency Span or Maximum Frequency Span without Option 70 Span Max = Stop – Start	Default Frequency Span or Maximum Frequency Span with Option 70 Span Max = Stop – Start
MS4642B	2 Hz	1.9990000000E+010 20 GHz – 10 MHz	1.9999930000E+010 20 GHz – 70 kHz
MS4644B	2 Hz	3.9990000000E+010 40 GHz – 10 MHz	3.9999930000E+010 40 GHz – 70 kHz
MS4645B	2 Hz	4.9990000000E+010 50 GHz – 10 MHz	4.9999930000E+010 50 GHz – 70 kHz
MS4647B	2 Hz	6.9990000000E+010 70 GHz – 10 MHz	6.9999930000E+010 70 GHz – 70 kHz

Standalone VNAs – Minimum Center and Maximum Center Frequencies

The minimum possible center frequency is the minimum start frequency plus 1 Hz. The maximum possible center frequency is the maximum stop frequency minus 1 Hz.

Table 1-8. Standalone VNAs – Minimum Center Frequency and Maximum Center Frequency

Model	Minimum Center Frequency without Option 70 Center Min = Start + 1 Hz	Minimum Center Frequency with Option 070 Center Min = Start + 1 Hz	Maximum Center Frequency All Models Center Max = Stop – 1 Hz
MS4642B	1.0000001000E+007 10 MHz + 1 Hz	7.0001000000E+004 70 kHz + 1 Hz	1.9999999999E+010 20 GHz – 1 Hz
MS4644B	1.0000001000E+007 10 MHz + 1 Hz	7.0001000000E+004 70 kHz + 1 Hz	3.9999999999E+010 40 GHz – 1 Hz
MS4645B	1.0000001000E+007 10 MHz + 1 Hz	7.0001000000E+004 70 kHz + 1 Hz	4.9999999999E+010 50 GHz – 1 Hz
MS4647B	1.0000001000E+007 10 MHz + 1 Hz	7.0001000000E+004 70 kHz + 1 Hz	6.9999999999E+010 70 GHz – 1 Hz

Standalone VNAs – Default Center Frequencies

The center frequency is equal to Start Frequency plus the Stop Frequency divided by two (2). The minimum possible frequency span is 2 Hz.

Table 1-9. Standalone VNAs – Default Center Frequencies

Model	Default Center Frequency without Option 70 Center Default = (Start + Stop)/2 Center Minimum Frequency Span = 2 Hz	Default Center Frequency with Option 70 Center Default = (Start + Stop)/2 Minimum Frequency Span = 2 Hz
MS4642B	1.0005000000E+010	1.0000035000E+010
MS4644B	2.0005000000E+010	2.0000035000E+010
MS4645B	2.5005000000E+010	2.5000035000E+010
MS4647B	3.5005000000E+010	3.5000035000E+010

VNA Systems with ME7828A Configured as Broadband System

For VNAs with ME7828A configured as a broadband system (mode selection and hardware selection):

- The VNA upper limit is 110 GHz
- The VNA lower limit is determined by the VNA installed option:
  - With Option 70, the lower limit is specified to 70 kHz, but is allowed to go to 40 kHz.
  - Without Option 70, the lower limit is 10 MHz.

VNA Systems with ME7828A Configured as a Millimeter-Wave System

For VNAs with ME7828A configured as a millimeter-wave system, the frequency range is established by the selected waveguide band. For example, 325 GHz to 500 GHz for WR-2.2.

VNA Systems with ME7828A Using Multiple Source

For VNAs with ME7828A configured using multiple sources, the frequency range is defined by the band equations and the installed hardware. The VNA system has no knowledge of the exact frequency range so it will report that of the base VNA.

VNA Systems with ME7838x Modular Broadband System

For VNAs equipped with the ME7838x modular broadband/millimeter-wave system in Modular BB mode:

- For ME7838A, The VNA acquires a broadband personality with an upper frequency limit of 125 GHz.
- For ME7838D, The VNA acquires a broadband personality with an upper frequency limit of 145 GHz.
- For ME7838E, The VNA acquires a broadband personality with an upper frequency limit of 110 GHz.
- The VNA lower limit is determined by the VNA installed option:
  - With Option 70, the lower limit is specified to 70 kHz, but is allowed to go to 40 kHz.
  - Without Option 70, the lower limit is 10 MHz.

VNA Systems with ME7838x Configured as a Millimeter-Wave System

For VNAs with ME7838x configured as a millimeter-wave system, the frequency range is established by the selected waveguide band. For example, 325 GHz to 500 GHz for WR-2.2.

Chapter 2 — Programming the VectorStar Series VNA

2-1 Introduction

This chapter provides an introduction to programming the VectorStar VNA with the SCPI programming language. It also includes descriptions of the command types that the MS4640B accepts, program command structures, data parameters and input/output specifications, and notational conventions. Information on the MS4640B status system and trigger system programming is also provided.

Note	When operating the VectorStar VNA through remote programming, the front panel user interface and controls are disabled. To return to local front panel control, press the front panel Clear/Tab key [**Clr -->|**], keyboard **Esc** key, or send the `RTL` command. For general information about GPIB, refer to Section 1-5 “IEEE 488 GPIB Description”.

2-2 Introduction to SCPI Programming

The Standard Commands for Programmable Instruments (SCPI) defines a set of standard programming commands for use by all SCPI compatible instruments. SCPI is intended to give the ATE user a consistent environment for program development. It does so by defining controller messages, instrument responses, and message formats for all SCPI compatible instruments. The IEEE 488 (GPIB) interface for the MS4640B is designed to conform to the requirements of SCPI 1999.0. The set of SCPI commands implemented by the MS4640B GPIB interface provides a comprehensive set of programming functions covering all of the major functions of the MS4640B.

Command Types

SCPI commands, which are also referred to as SCPI instructions, are messages to the instrument to perform specific tasks. The MS4640B command set, introduced in this chapter, includes these command types:

- “IEEE 488.2 Commands”
- “System Commands”
- “SCPI Commands”
- “Native SCPI Commands”
- “Anritsu Lightning VNA Commands”
- “HP8510 Commands”

2-3 IEEE 488.2 Commands

The IEEE-488.2 commands are defined in the IEEE-488.2 standard and must be implemented by all SCPI compatible instruments. The mandated commands listed in Table 2-1 are identified by the asterisk (*) at the beginning of the command keyword. These commands are used to control instrument status registers, status reporting, synchronization, and other common functions. The IEEE 488.2 required common commands are described in detail in the first half of Chapter 3, “IEEE Commands” starting with “IEEE 488.2 Commands” on page 3-2.

Table 2-1. IEEE 488.2 Mandated Commands

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

2-4 System Commands

The set of system commands are primarily used to control the state of the MS4640B for system diagnostics, hardware calibration, and troubleshooting.

2-5 SCPI Commands

There are two general classifications of SCPI commands described in the two sections below. They are:

- Required (or mandated) SCPI Commands
- Native SCPI Commands

Note that the Required SCPI Commands are a subset of the Native SCPI commands.

Required SCPI Commands

The required SCPI commands are listed in the table below

Table 2-2. SCPI Required or Mandated Commands

:STATus
  :OPERation
    [:EVENt]?
    :CONDition?
    :ENABle
  :QUEStionable
    [:EVENt]?
    :CONDition?
    :ENABle
	:SYSTem
	  :ERRor
	    [:NEXT]?

The SCPI Required Commands are described in detail in Chapter 5, “SCPI Commands” in the following sections:

- “:STATus:OPERation Subsystem” on page 5-531
- “:STATus:QUEStionable Subsystem” on page 5-533

- “:SYSTem Subsystem” on page 5-536

Native SCPI Commands

The majority of the commands are native SCPI commands and are also described in detail in Chapter 5, “SCPI Commands”.

The commands are organized into subsystems, starting with the “:CALCulate{1-16}:APPLication:MEASurement Subsystem” on page 5-13 and ending with the “:TRIGger[:SEQuence] Subsystem” on page 5-552.

Depending on the number of keywords in the command, the subsystems are grouped by either the first two keywords (such as :CALCulate:MARKer Subsystem) or the first three keywords (such as :CALCulate{1-16}[:SELected]:CONVersion Subsystem)

The commands are listed in strict ASCII sort sequence.

See the sections below starting with “Command Requirements” on page 2-5 for definitions of parameters and other notations.

2-6 Anritsu Lightning VNA Commands

Many (but not all) commands used on the Anritsu Lightning 37xxxD/E VNAs are supported on the MS4640B Series VNA. The supported and non-supported commands are described in the companion manual, VectorStar MS464xB Series VNA Programming Manual Supplement – 10410-00323.

For additional detailed information on Lightning 37xxxx VNA commands and programming the Anritsu Lightning 37xxxD/E VNAs, refer to either of the following manuals available on the Anritsu web site:

- Lightning 37xxxD Programming Manual – 10410-00262
- Lightning 37xxxE Programming Manual – 10410-00301

Using Anritsu Lightning VNA Commands

All Anritsu Lightning VNA commands operate on the VectorStar MS4640B VNA Active Channel, and there are no Lightning commands which can change the VectorStar VNA active channel to another one. If the VectorStar VNA is configured with multiple channels, the Lightning commands will then only operate on the currently active VectorStar channel. No error will be generated. Other Lightning command limitations are noted in the command listing in the Programming Manual Supplement – 10410-00323. See the sections below for definitions of parameters and other notations.

VectorStar VNA Language Set to Lightning

When using Anritsu Lightning commands, note that:

1. Recognition of the Lightning command set is provided for compatibility with existing Lightning ATE programs, and the use of the Lightning command set is not recommended for new development.
2. Some of the Lightning commands may not work as expected if the programming Language is not set to Lightning via the LANG command (or LANG LIGHT).
    - For example, markers in the Native language are trace based. This means that each trace has its own set of markers, independent of the other traces.
    - In Lightning there are only 6 markers. If you move Marker1 on Trace1 to 3 GHz, Marker1 on the other 3 traces will also go to 3 GHz.

If you want the Lightning behavior on the markers, you need to set the Language to Lightning.

2-7 HP8510 Commands

Many commands used on the HP8510 VNA are also supported on the MS464xB Series VNA. The supported commands are described in detail in the aforementioned Programming Manual Supplement – 10410-00323. See the sections below for definitions of parameters and other notations.

2-8 Command Requirements

Query Commands

All commands, unless specifically noted in the commands syntax descriptions, have a query form. Exceptions are noted as:

- Commands without a query form have a query status of "No Query".
- Queries without a command form have a query status of "Query Only".

As defined in IEEE-488.2, a query is a command with a question mark symbol appended (examples are `*ESR?` and `*TST?`). When a query form of a command is received, the current setting associated with the command is placed in the output buffer. Query commands always return the short form of the parameter. For example, `NORMal` or `INVerted` is returned as `NORM` or `INV`. Boolean values are returned as `1` or `0`, even when they can be set as `ON` or `OFF`.

Command Names

Typical SCPI commands consist of one or more keywords, parameters, and punctuation. SCPI command keywords can be a mixture of upper and lower case characters. Except for common commands, each keyword has a long and a short form. In this manual, the long form is presented with the short form in upper case and the remainder in lower case. For example, the long form of the command keyword to control the instrument display is `:DISPlay`.

The short form keyword is usually the first four characters of the long form (example: DISP for DISPlay). The exception to this is when the long form is longer than four characters and the fourth character is a vowel. In such cases, the vowel is dropped and the short form becomes the first three characters of the long form. Example: the short form of the keyword `:POWer` is `:POW`.

Some command keywords may have a numeric suffix to differentiate between multiple instrument features such as multiple pulse widths. For example, keywords `:WIDTh2` (or `:WIDT2`).

As with any programming language, the exact command keywords and command syntax must be used. The syntax of the individual commands is described in detail in Chapter 5, "SCPI Commands". Unrecognized versions of long form or short form commands, or improper syntax, will generate an error.

Hierarchical Command Structure

All SCPI commands, except the common commands, are organized in a hierarchical structure similar to the inverted tree file structure used in most computers. The SCPI standard refers to this structure as "the Command Tree." The command keywords that correspond to the major instrument control functions are located at the top of the command tree. The command keywords for the MS4640B SCPI command set are shown in the diagram below.

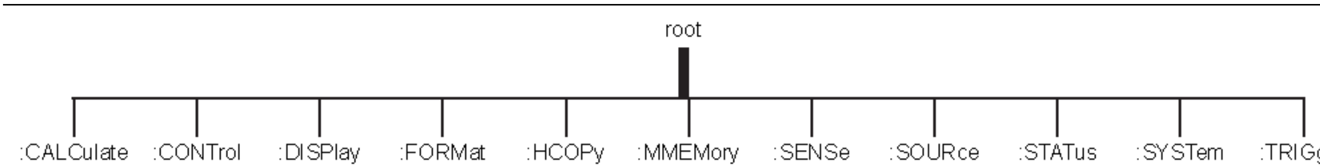


Figure 2-1. MS4640B Partial SCPI Command Tree

All MS4640B SCPI commands have one or more subcommands (keywords) associated with them to further define the instrument function to be controlled. The subcommand keywords may also have one or more associated subcommands (keywords). Each subcommand level adds another layer to the command tree. The command keyword and its associated subcommand keywords form a portion of the command tree called a command subsystem. The `:FORMat` command subsystem is shown below.

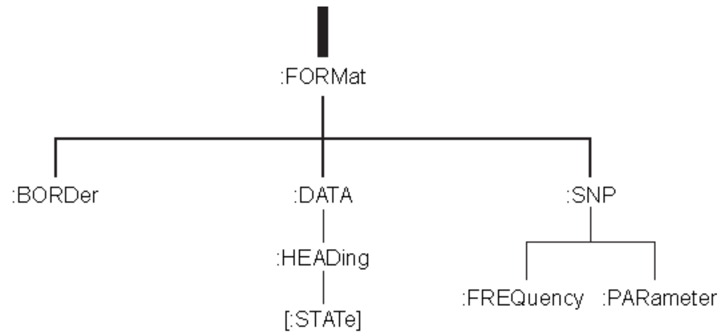


Figure 2-2. SCPI `:FORMat` Subsystem

Data Parameters

Data parameters, referred to simply as "parameters," are the quantitative values used as arguments for the command keywords. The parameter type associated with a particular SCPI command is determined by the type of information required to control the particular instrument function. For example, Boolean (ON | OFF) type parameters are used with commands that control switch functions.

The command descriptions in Chapter 5, "SCPI Commands" specify the type of data parameter to be used with each command. The most commonly used parameter types are numeric, extended numeric, discrete, and Boolean.

- **Numeric**

  Numeric parameters comprise integer numbers or any number in decimal or scientific notation, and may include polarity signs. This includes <NR1>, <NR2>, and <NR3> numeric data as defined in "Parameter Notations" on page 2-8. This type of numeric element is abbreviated as <NRf> throughout this document.

- **Extended Numeric**

  Extended numeric parameters include values such as MAXimum and MINimum. Extended numerics are not supported in the current MS4640B SCPI implementation.

- **Discrete**

  Discrete parameters, such as INTernal and EXTernal, are used to control program settings to a predetermined finite value or condition.

- **Boolean**

  Boolean parameters represent binary conditions and may be expressed as ON, OFF or 1, 0.

Note	The VectorStar command parser will generally accept all numerical values within the parameter ranges specified. In cases where a command parameter value is outside of the indicated range or resolution of the instrument, the nearest appropriate value will be entered.

2-9 Notational Conventions

The SCPI interface standardizes command syntax and style that simplifies the task of programming across a wide range of instrumentation. As with any programming language, the exact command keywords and command syntax must be used. Unrecognized commands or improper syntax will not function.

General Notations

The syntax conventions that are used for all SCPI command keywords and data parameter descriptions in this manual are described below:

Table 2-3. General Notations

:	A colon links command keywords together to form commands. The colon is not an actual part of the keyword, but is a signal to the SCPI interface parser. A colon must precede a root keyword immediately following a semicolon (see “Notational Examples” on page 2-9).
;	A semicolon separates commands if multiple commands are placed on a single program line (see “Notational Examples” on page 2-9).
[ ]	Square brackets enclose one or more optional keywords.
{ }	Braces enclose one or more keyword parameters that may be included one or more times.
|	A vertical bar (also called a “pipe”) indicates “or” and is used to separate alternative parameter options. For Example: ON | OFF is the same as ON or OFF.
< >	Angle brackets enclose parameter descriptions.
::=	Means “is defined as” For example: <a>::=<b><c> indicates that <b><c> can replace <a>.

For further information about SCPI command syntax and style, refer to the **Standard Commands for Programmable Instruments (SCPI) 1999.0 document**.

Parameter Notations

The following syntax conventions are used for all data parameter descriptions in this manual:

Table 2-4. Parameter Notations

Parameter	Definition
<ASCII>	A non-delimited 7-bit ASCII text. The end of the text must be terminated with the 0A character (decimal 10) and concurrent setting (^) of the GPIB End of Transmission State (EOI). <ASCII> (also called <Arbitrary ASCII>) text is transmitted only at the end of a program or response message.
<block>	IEEE-488.2 block data format. Can be in ASCII, XML, or other format.
<bNR1>	Boolean values in <NR1> format; numeric 1 or 0
<boolean>	ON | OFF. Can also be represented as 1 or 0, where 1 means ON and 0 means OFF Boolean parameters are always returned as 1 or 0 in <NR1> format by query commands
<char>	<CHARACTER PROGRAM DATA> Examples: CW, FIXed, UP, and DOWN
<INF>	Positive Infinity. Positive infinity is represented as 9.9E37. The numeric value for positive infinity fits into a 32-bit IEEE 754 floating point number.
<integer>	An unsigned integer without a decimal point (implied radix point)
<NA>	Not Applicable
<NAN>	Not A Number. Not a number is represented as 9.91E37 and is defined in IEEE 754. Typically used where applications are dividing zero by zero or subtracting infinity from infinity. NAN is also used to represent missing data such as a trace that has not been yet acquired.
<NINF>	Negative Infinity. Negative infinity is represented as -9.9E37. The numeric value for negative infinity fits into a 32-bit IEEE 754 floating point number.
<NR1>	A signed integer without a decimal point (implied radix point)
<NR2>	A signed number with an explicit radix point
<NR3>	A scaled explicit decimal point numeric value with an exponent (e.g., floating point number)
<NRf>	Values in NR1, NR2, or NR3 formats are accepted. Logically, <NR1> | <NR2> | <NR3>
<numeric_value>	Also <nv> or SCPI numeric value as: <NRf> | MINimum | MAXimum | DEFault | UP | DOWN | NAN or NotANumber | INF or INFinity | NINF or NegativeINFinity or other types
<nv>	<numeric_value> or SCPI numeric value as: <NRf> | MINimum | MAXimum | DEFault | UP | DOWN | NAN or NotANumber | INF or INFinity | NINF or NegativeINFinity or other types
<string>	<STRING PROGRAM DATA> ASCII characters surrounded by double quotes For example: “C:\Anritsu\VectorStar\filename.s2p”
MPND	Numeric Limit. Maximum Positive/Negative Double Precision Number. ± 1.792 693 134 860 E+308
MPNF	Numeric Limit. Maximum Positive/Negative Float Number ± 3.402 819 E+38
MPNI	Numeric Limit. Maximum Positive/Negative Integer - 2 147 483 648 to +2 147 483 647

Refer to “Data Transmission Methods” on page 2-11 for detailed information about parameter input/output and transferring data to/from the instrument.

Notational Examples

The following is an example showing command syntax:

```
:SENSe1:FREQuency:STARt 2.0E9
```

Command statements read from left to right and from top to bottom. In the command statement above, the :FREQuency keyword immediately follows the :SENSe1 keyword with no separating space. A space is required between the command string and its argument.

Note that the first keyword in the command string does not require a leading colon; however, it is good practice to always use a leading colon for all keywords.

The following is an example of a multiple command statement that uses two separate commands in a single statement:

```
:SENSe1:FREQuency:STARt 2.0E9;:SENSe1:FREQuency:STOP 20.0E9
```

Using the command keyword short form, the command string above would be:

```
:SENS1:FREQ:STAR 2.0E9;:SENS1:FREQ:STOP 20.0E9
```

Note the semicolon used to join the commands. Also note the leading colon used immediately after the semicolon.

Band Equation

MPNI, Limited by the band equation. The band equation depends whether or not CW is set.

If CW is off, the band equation equals:

```
Source = (Multiplier/Divisor) x (Frequency plus Offset Frequency)
```

If CW is on, the band equation equals:

```
Source = (Multiplier/Divisor) + Offset Frequency
```

2-10 Numeric Data Suffix Reference

Unit suffixes are not required for data parameters, provided the values are scaled for the global default units. The MS4640B SCPI default units are:

- Hz (hertz) for frequency-related parameters
- s (second) for time-related parameters
- m (meter) for distance-related parameters
- ohm for impedance-related parameters
- dB for power-related parameters
- Henry and Farad for reactance-related parameters

For example, the commands below set the MS4640B marker 1 frequency to 3 GHz.

```
:CALCulate1:MARKer1:X 3000000000
:CALCulate1:MARKer1:X 3.0E9
```

The following table provides a reference to the I/O parameter types (and the appropriate multiplier) used with the MS4640B VNA.

Table 2-5. Numeric Data Suffix

Code	Parameter Type	Multiplier
DB, DBL, DBM	Power	1.0
DEG	Phase	1.0
RAD	Phase	$180/\pi$ (180/Pi)
HZ	Frequency (Hertz)	1.0
KHZ	Frequency (Kilohertz)	1.0E3
MHZ	Frequency (Megahertz)	1.0E6
GHZ	Frequency (Gigahertz)	1.0E9
REU	Real	1.0
IMU	Imaginary	1.0
S	Time	1.0
MS	Time (Millisecond)	1.0E-3
US, USC	Time (Microsecond)	1.0E-6
NS, NSC	Time (Nanosecond)	1.0E-9
PS, PSC	Time (Picosecond)	1.0E-12
M, MTR	Distance (Meter)	1.0
CM, CMT	Distance (Centimeter)	1.0E-2
MM, MMT	Distance (Millimeter)	1.0E-3
OHM	Impedance	1.0
V, VLT	Voltage	1.0
MV	Voltage (Millivolt)	1.0E-3
XM3	Unitless	1.0E-3
XX1	Unitless	1.0
XX3	Unitless	1.0E3

2-11 Data Transmission Methods

Data transmissions to and from the MS4640B conform to the protocols specified by the IEEE 488.2 GPIB Standard. The 488.2 Standard specifies how any data, such as ASCII numbers, strings, or blocks of data bytes, will be transmitted over the GPIB. This section describes the various transmission methods in use by the MS4640B. The transmission method names described below (also called notations) will be used throughout the Programming Manual when describing specific MS4640B data transfer commands. Data transmission notations are easily distinguished in text as they are always shown surrounded by the “less than” and the “greater than” characters (< >). The transmission type notations used in describing various MS4640B VNA data transmissions are:

- For ASCII numbers, the notations are: <NR1>, <NR2>, <NR3>, or <NRf>
- For ASCII strings (printable characters and print formatting codes), the notation is: <string>
- For generic (7-bit) ASCII characters, the notation is: <Arbitrary ASCII>
- For generic binary bytes, (7-bit ASCII or binary), the notation is: <block>

<NR1>

This notation represents ASCII integer values. A comma (,) is used to separate multiple values sent in a single command's input or output string. Examples of values that can be represented by <NR1> notation:

```
10
-29,179
```

<NR2>

This notation represents ASCII floating point values in decimal point format. A comma (,) is used to separate multiple values sent in a single command's input or output string. Examples of values that can be represented by <NR2> notation:

```
1.0
-0.00015
12.743,-180.07
```

<NR3>

This notation represents ASCII floating point values in exponential format (scientific notation). A comma (,) is used to separate multiple values sent in a single command's input or output string. Examples of values that can be represented by <NR3> notation:

```
1.0E9
-7.056E3
9.0E-2,3.42E2
```

<NRf>

This notation is used to signify that data can be in either <NR1>, <NR2>, or <NR3> format as described above. Examples of values that can be represented by <NRf> notation:

```
1.0E-9
10.005
-83,4.5E2,-234.9901
```

<string>

This notation represents a string of ASCII characters (including non-printable characters) that is delimited (surrounded) with either single quotes (' ') or double quotes (" "). The string can include text formatting characters such as line feed, space, carriage return, or printer control characters. Note that if a double quote character must be sent as part of the string, then it must be followed by an additional double quote. Alternatively, the string can be sent using single quotes (See "cal_file" example below). Examples of data represented by <string> notation:

```
"1/15/98"
"Save ""cal_file"" now"
'Save "cal_file" now'
```

<ASCII> or <Arbitrary ASCII>

This notation represents undelimited 7-bit ASCII text. The end of the text must be terminated with the 0A character (decimal 10) and concurrent setting (^) of the GPIB End of Transmission State (EOI). This requirement makes it necessary for <Arbitrary ASCII> text to be transmitted only at the end of a program or response message, i.e., at the end of a multiple input or output statement. Example of data represented by <Arbitrary ASCII> notation:

```
ANRITSU,MS4642B,123456,1.0<0A^EOI>
```

The example shows a sample response from the *IDN?, 488.2 common query. In the example, the instrument identifies itself as an ANRITSU MS4642B, with serial number 123456, and software version 1.0 installed.

Note	Note that decimal 10 (0A character) must be sent with the EOI to signal the end of transmission.

<block> or <arbitrary block>

This notation represents data that is transmitted as 8-bit data bytes (00–FF hex, 0–255 decimal, notation is <DAB>). This is useful for transmitting large blocks of:

- Formatted ASCII data
- Formatted XML data
- Formatted binary data
- Unformatted binary data

The data stream is immediately preceded by a variable length ASCII header that is encoded with the number of data bytes to be sent. The header always starts with the pound (#) character. The header and the transmitted data messages are described as follows:

```
#nm1..mn<DAB1>...<DABm>
```

Where:

= The pound sign character. Required for binary data transfer.

n = Number of digits to follow (m1..mn) that make up the number m.

m1..mn = Taken together, this makes up the number m which is the number of data bytes to follow that constitute the requested data.

<DAB> = An 8 bit binary data byte. This is the data (or information) being sent.

Note	If n = 0, then m is omitted, and transmission end is signaled by sending the line feed character (0A, or decimal 10) and concurrent setting (^) of the GPIB End Of Transmission State (EOI) immediately following the last <DAB>.

Example 1: #3204<DAB1>...<DAB204>

Example 1 shows how 204 8-bit bytes are transmitted using the proper header. The header in this example is comprised of 5 characters (#3204). It begins with the pound character (#). The next character (3) indicates there are 3 digits to follow that indicate the number of bytes being transmitted (204). The next three characters (204) indicate the number of data bytes being transmitted immediately after the header. Next comes the actual data bytes, or information, being transmitted (<DAB1>...<DAB204>).

Example 2: #512808<DAB1>...<DAB12808>

Example 2 shows how 12808 bytes are transmitted using the proper header. The header in this example is comprised of 7 characters (#512808). It begins with the pound character (#). The next character (5) indicates there are 5 digits to follow that indicate the number of bytes being transmitted (12808). The next five characters (12808) indicate the number of data bytes being transmitted immediately after the header. Next comes the actual data bytes, or information, being transmitted (<DAB1>...<DAB12808>).

Note	Examples 1 and 2 above demonstrate the <block> form referred to as <Definite Length Arbitrary Block>. It is so called because the number of data bytes being transmitted is known from the encoded header.

Example 3: #0<DAB1>...<DABm><0A^EOI>

Example 3 shows how an unknown number of bytes are transmitted using the proper header. The header in this example is comprised of 2 characters (#0). As usual, the header begins with the pound character (#). The next character (0) indicates there is an unknown number of data bytes being transmitted immediately after the header. Next comes the actual data bytes being transmitted (<DAB1>...<DABm>). The end of the data stream is signaled by sending the line feed character (0A, or decimal 10) and concurrent setting (^) of the GPIB End of Transmission State (EOI).

Note	Example 3, above, demonstrates a special form of <block> data referred to as the <Indefinite Length Arbitrary Block>. It is so called because the number of data bytes being transmitted is unknown, and therefore cannot be encoded in the header. Instead, the header always consists of the pound and zero characters (#0) and end of the data stream is always signaled by sending the line feed character (0A, or decimal 10) and concurrent setting (^) of the GPIB End of Transmission State (EOI). This requirement makes it necessary for <Indefinite Length Arbitrary Block> text to be transmitted only at the end of a program or response message (at the end of a multiple input or output statement).

<char>

Character program data such as CW, FIXed, UP, and DOWN. A single instance in a command or query is <char>. If multiple instances are required, each is identified such as <char1> or <char2> and the individual elements are separated by commas:

- <char1>,<char2>
- <char1>,<char2>,<char3>
- <char1>,<char2>,<char3>,<char4>

MPND

The instrument numeric limit as the Maximum Positive/Negative Double Precision Number or:

```
+/- 1.792 639 134 86 E+308
```

MPNF

The instrument numeric limit as the Maximum Positive/Negative Float Number or:

```
+/- 3.402 819 E+38
```

MPNI

The instrument numeric limit as the Maximum Positive/Negative Integer or:

```
- 2 147 483 648 to +2 147 483 647
```

Formatting Data Output

Three commands are provided to alter the way the arbitrary block header for output data is formed.

- **FDH0**

  Specifies that the length of the arbitrary block header will be minimized; that is, the byte count section will not contain leading zeros, thus its length is indeterminate. This means that a program must decode the header in order to skip over it. FDH0 is the default mode when the programming language is selected as Lightning (see “Configuring the Remote Language” on page 1-21).

- **FDH1**

  Specifies that the length of the arbitrary block header will be fixed at 11 characters. This is accomplished by forcing leading zeros as required in the byte count section. This means that a program can skip over the arbitrary block header by skipping 11 characters. FDH1 is the default mode when the programming language is selected as Native (see “Configuring the Remote Language” on page 1-21).

- **FDH2**

  Specifies that no arbitrary block header will be sent with the next transmission. This mode is not in compliance with IEEE 488.2 specifications and will persist for all subsequent program messages.

- **FDHX?**

  FDH mode query yields the following results:

  - 0: FDH0
  - 1: FDH1
  - 2: FDH2

ASCII or Binary Data Format

The following sections discuss the various data output formats:

Non-Array Data

The formats used for data transfers not involving numerical data arrays are preset. They always occur in either binary format or ASCII format, depending on the data. These data transfers include a variety of information. Examples include:

- Instrument setup strings
- Marker data, queries
- Disk directory listings

Numerical Data Arrays

Numerical data array transfers are used to transfer the following types of data:

- Measurement data
- Calibration data
- Sweep frequency, time, or distance values

Each of these data transfer types are individually explained below. You can select either binary or ASCII format for data transfers involving numerical data arrays. The commands described below select and keep the format for all subsequent data transfers.

- **ASCII Format - FMA**

  ASCII formatted values represented in <NR1>, <NR2>, <NR3>, or <NRf> formats. The MS4640B VNA accepts any of the above formats as input. It will always output values using <NR3> exponential format with each value represented using 18 characters plus a comma to separate multiple values.

- **Binary Formats**
  - **FMB**

    Each eight consecutive data bytes represent one floating point value in IEEE 754 64-bit format (double precision, 8 byte, floating point value).

  - **FMC**

    Each four consecutive data bytes represent one floating point value in IEEE 754 32-bit format (single precision, 4 byte, floating point value).

  - **FMX?**

    FMA, FMB, FMC format selection query

- **Byte Ordering**
  - **MSB**

    Byte ordering is most significant byte first. For use only with FMB and FMC. This is the optional byte mode for the MS4640B.

  - **LSB**

    Byte ordering is least significant byte first. For use with FMB and FMC. This is required for transferring data to/from Intel/IBM based computers. LSB is the default mode.

- **XSB?**

  MSB, LSB format selection query.

- **FMT0**

  Turn ASCII enhancement off (normal default mode).

- **FMT1**

  Turn ASCII enhancement on.

- **FMTX?**

ASCII enhancement ON/OFF status query.

The following SCPI commands select either ASCII or Binary format as described above:

`:FORMat:DATA <char>`

Where the **<char>** arguments of ASC or ASCII ::= FMA, REAL ::= FMB, REAL32 ::= FMC

`:FORMat:DATA?` is the ASC or ASCII, REAL, or REAL32 format selection query.

`:FORMat:BORDer <char>`

Where the **<char>** arguments of NORMAL ::= MSB, SWAPPED ::= LSB

`:FORMat:BORDer?` is the MSB | LSB format selection query.

Enhanced ASCII Formatting

Enhanced ASCII formatting can be applied to both non-array ASCII data and numerical data arrays in the FMA format when this data is output within an <block> format. The format selectively replaces comma data element separators with a line feeds (ASCII 10) in order to enhance the visual effect. The following provides two examples of this enhanced structure:

- An unenhanced directory listing

```
#9000000392Directory of C:\ 1-30-96 13:03,UTIL <DIR> 1-25-96 12:58,PLOT
BMB 38462 1-22-96 14:41,PLOT BMC 307446 1-22-96 14:41,TTT CAL
44174 1-22-96 17:02,TTT2 CAL 44174 1-22-96 17:16,PLOT1 DAT
10323 1-22-96 14:03,PLOT1 HGL 19899 1-22-96 14:02,PLOT2 HGL
38462 1-25-96 13:16,8 Files 502940 Bytes
```

- An enhanced directory listing

```
#9000000392
Directory of C:\ 1-30-96 13:03
UTIL <DIR> 1-25-96 12:58
PLOT BMB 38462 1-22-96 14:41
PLOT BMC 307446 1-22-96 14:41
TTT CAL 44174 1-22-96 17:02
TTT2 CAL 44174 1-22-96 17:16
PLOT1 DAT 10323 1-22-96 14:03
PLOT1 HGL 19899 1-22-96 14:02
PLOT2 HGL 38462 1-25-96 13:16
8 Files 502940 Bytes
```

- An unenhanced response to OCD

```
#9000000189-9.99750733376E-01, 3.21409821510E-01, 3.60706359148E-01,
9.82860028744E-01, 7.76742696762E-01,-5.06587028503E-01,-5.07535457611E-01,
-8.45697641373E-01,-6.10321164131E-01,6.05827927589E-01
```

- An enhanced response to OCD

```
#9000000189
-9.99750733376E-01, 3.21409821510E-01
3.60706359148E-01, 9.82860028744E-01
7.76742696762E-01,-5.06587028503E-01
-5.07535457611E-01,-8.45697641373E-01
-6.10321164131E-01, 6.05827927589E-01
```

2-12 Calculating the Byte Size

This section describes the factors for calculating the byte size of responses to selected remote-only queries. The byte size of the resultant data from several of the remote only queries depends on several factors:

- Parameters per Output
- Numbers Output per Data Point
- Bytes Output per Number
- Size of Block Data
- Number of Bytes Output

Numbers Output-per-Data Point (NODP)

The data for each data point is a complex number (A + jB) where A and B are floating point numbers. This data is saved internally for use and possible future output. Additionally, if an RF correction is active, the RF correction is applied to the RAW measurement and the result is saved internally for use and possible future output. Either the RAW or CORRECTED data are taken and converted into the data format for the display type selected.

This data is saved internally in the FORMATTED (final) measurement form for use and possible future output. When this conversion takes place, the data will, in most cases, still be two orthogonal numbers. However, several of the displays types throw away a portion of the data and the result will be one number only. The display types that produce only one number are:

- Group Delay
- Imaginary
- Linear Magnitude
- Log Magnitude
- Phase
- Power Out
- Real
- SWR

To summarize, the RAW, CORRECTED, and FORMATTED data output will be two numbers-per-point, unless the display type is one of those mentioned above.

Bytes Output-per-Number (BOPN)

The number of bytes output per number is shown below:

Table 2-6. Bytes Output per Number

Number	Output Format	Output-per-Number
FMA	(ASCII)	14 plus comma (short form data) 19 plus comma (long form data)
FMB	(double precision binary)	8
FMC	(single precision binary)	4

Size of Data Block (SODB)

In the case where there is only one parameter to output, the formula is:

```
SODB = NODP * BOPN * Number of points in the sweep
```

If the command is O4SC, O4FD, or O4SR, the formula is:

```
SODB = 8 * BOPN * Number of points in the sweep
```

Number of Bytes Output (NBO)

The number of bytes output is the number of bytes transmitted over the GPIB. In most cases, the data block is proceeded by an arbitrary block header followed by an end character (line feed), as shown below:

- Response Message = [Arbitrary Block Header] + [Data Block] + [End Character]

The size of the end character is one byte. The size of the arbitrary block header is variable between 2 and 11. If we always assume an arbitrary block header size of 11, then: NBO = 12 + SODB. For example:

- The VNA is set up for a one channel, four-trace display with a 1601 point sweep.
- Trace 1 is displaying S11 in LogMag and Phase format
- Trace 2 is displaying S12 in LogMag format
- Trace 3 is displaying S21 in Phase format
- Trace 4 is displaying S22 in Smith Chart format
- The output formatting commands CH2, FMC, and LSB are received

The number of output bytes for the O4FD query command is:

```
NBO = 12 + 8 * 4 * 1601 = 51244 bytes
```

The number of output bytes for the ORD query command is:

```
NBO = 12 + 2 * 4 * 1601 = 12820 bytes
```

The number of output bytes for the OFD3 query command is:

```
NBO = 12 + 1 * 4 * 1601 = 6416 bytes
```

The number of output bytes for the FMA or O4SR query command is:

```
NBO = 12 + 8 * 19 * 1601 = 243364 bytes
```

2-13 GPIB Input Buffer Size and NRFD Holdoff

VectorStar VNAs provide a very large input buffer that can hold up to 100 commands. Each command plus any associated data can be as large as the amount of memory available in the VNA at the time. If a programmer attempts to exceed 100 commands, the GPIB bus will go into Not Ready For Data (NRFD) Holdoff. This Holdoff condition will hold onto the controller PC until a command is executed which frees up room for another command. Then the new command will be read in. Some controller PCs can detect this Holdoff condition and programmers interpret this condition as an Error. It is not an Error. Rather, it is a function provided by any listener device to guarantee that commands and data are not lost. It might be an ATE program error, but not a VNA error.

2-14 Synchronization of GPIB Commands

The VectorStar VNA provides synchronization of the GPIB by executing commands in a serial fashion. Subsequent commands will not be parsed and executed until the current GPIB command is parsed and completely executed. Indeed, as far as VNA operations are concerned, if this serial execution method is not incorporated, the VNA, the GPIB bus, and the controller PC will be in chaos. Avoiding this chaos condition is so important that the IEEE488.2 standard mandated 3 commands to be provided: *OPC, *OPC?, and *WAI.

Note	For more information, see the descriptions of the *OPC, *OPC?, and *WAI commands in Chapter 3, “IEEE Commands”.

This synchronization is accomplished by the GPIB parser which waits for a Completed signal from the internal interface after starting execution of the command. While the parser is waiting, it is not parsing newer commands. Subsequent commands are put into the input buffer, awaiting their turn to be parsed and executed.

2-15 Forcing the Parser to Stop Waiting

The parser will wait forever for the Completed signal as discussed above in “Synchronization of GPIB Commands”. Therefore there is no GPIB command (which would itself require parsing) that can stop the parser from waiting. The controller PC will have to send a GPIB bus command called Device Clear (DCL) or Selected Device Clear (SDC). SDC is directed at a particular device address. DCL will perform the same action on all devices on the Bus. Among the required things DCL and SDC do is reset the Parser. This causes the parser to stop waiting for the ‘Completed’ signal and get ready to execute a command. In the **IEEE488.2 Specification**, see **IEEE488.2 Section 5.8** and **IEEE488.1 Section 4.10** for a discussion of DCL and SDC.

2-16 How Can I Abort an RF or Hardware Calibration

Flat Test Port Power Calibration and Linearity Calibrations are also Hardware Calibrations. From the front panel, an operator can start and abort these calibrations with ease. Usually there is a STOP or ABORT button. One can also send the GPIB command ‘ABORT’ to abort these manually initiated calibrations. However, when a calibration is initiated from the GPIB, the parser is busy waiting for the ‘Completed’ message from the internal interface and is not available to parse the ‘ABORT’ command to abort the calibration. Refer to the two sections above ‘Synchronization of GPIB commands’ and ‘Forcing the Parser to stop waiting’ to see how that is done. Once the parser is ready for a new command, send the ‘ABORT’ command.

2-17 GPIB Time-Out Settings

VectorStar and Lightning VNAs provide synchronization with the GPIB by executing commands in a serial fashion. A new command will not be parsed and executed until the previous command has finished processing. Therefore, the synchronization commands stipulated in IEEE 488.2 (`*OPC?`, `*OPC`, and `*WAI`) execute with ease.

Lightning VNAs provide a very large input buffer that permits storing many commands until they can be executed. The VectorStar VNAs can store very large command strings (a series of characters terminated with a line feed), but only two strings at most. This creates a situation where the GPIB bus can go into Not Ready for Data (NRFD) hold off if the controller sends more than two command strings at a time. NRFD hold off will hold onto the controller until it can store the data byte that is currently being transferred. This hold off merely guarantees that no data byte will be lost from a message. Although some controllers can detect this hold off, it's occurrence is of no consequence and it is how instruments are kept communicating at the same rate on the GPIB.

Some commands may take a very long time to execute, such as waiting for the end of a sweep when the IF bandwidth is very low (1 Hz to 100 Hz). In spite of the long sweep time, the controller should avoid timing out because it creates a situation where synchronization and data can be lost. A controller time-out also leaves the GPIB and the VNA in an unknown I/O state. This unknown I/O state may not be responsive because the proper data handshake has been interrupted, thus creating a hung bus.

In many cases, the GPIB parser will also be busy participating in a long event, such as when the commands `TRS;WFS` are sent. The parser will not be available until the sweep is finished. The only way to get the parser to stop its current task and start processing new commands is to assert the Interface Clear Line (`IFC`) and issue a GPIB bus command called Device Clear (`DCL`). These two commands together brings the parser back to the Ready for Data (RFD) state. The commands also reset the input and output buffers, which results in both input and output data being lost.

Below is a sequence that shows how to apply a time-out properly:

```
SETTIMEOUT(40000); // Sets a longer time-out for the controller.
OUTPUT ; RST // RST (reset) is a command that can take 30 seconds or longer.
ENTER ; ONP // Outputs the number of points. The actual wait occurs here.
SETTIMEOUT(20000); // Sets the time-out back to its normal value.
```

Setting the proper time-out on the controller is very important to guarantee that the GPIB will be synchronized and data will not be lost. One should choose a time-out that allows most operations to finish without problems and set the time-out to different values on those commands that require a longer time to execute. If time-outs do occur in program execution, the time-out settings for the particular commands in question should be increased. If a time-out is due to an application error, such as sending a syntax error preceding a query or an impossible state is set up such as being in HOLD and waiting for the end of the sweep, the coding error should be fixed and the time-out setting left as it is. Timing out also leaves the bus in an unknown I/O state, so a DCL should be sent to synchronize handshaking. USB or VXI-11 do not provide an IFC, but they do provide a DCL.

2-18 Trace Type Parameters and Coefficients

The following table provides a reference for the various graph types and related data types used in the VectorStar Series VNA.

Table 2-7. Trace Parameters and Coefficients (1 of 4)

Trace Name SCPI Keyword Display Trace Abbreviation	Trace Graph Format	Default Reference Level	Reference Level Range	Default Resolution	Resolution Range Parameter / Division	Default Scale Position	Scale Reference Range - Num of VertIcal Div.
Group Delay **GDELay** **GDEL**	Single Rectilinear Graph	0	±9.9999E2	1 microsecond	1E-13 to 1E9	5	4 to 30
Imaginary **IMAGinary** **IMAG**	Single Rectilinear Graph	0	±9.9999E2	1 Unit (U)	1E-5 to 1E6	5	4 to 30
Linear Mag and Phase **LINPHase** **LINPH**	Double Rectilinear Graphs	Top=0 Bottom = 0	±9.9999E2	Top = 10 U Bottom = 45 degrees	-NA-	Top = 5 Bottom = 5	Top = 4 to 30 Bottom = 4 to 30
Log Mag and Phase **LOGPHase** **LOGPH**	Double Rectilinear Graphs	Top=0 Bottom = 0	±9.9999E2	Top = 10 dB Bottom = 45 degrees	Top = 1E-3 to 1E3 Bottom = 1E-2 to 1E6	Top = 5 Bottom = 5	Top = 4 to 30 Bottom = 4 to 30
Linear Mag **MLINear** **MLIN**	Single Rectilinear Graph	0	±9.9999E2	10 U	1E-5 to 1E6	5	4 to 30
Log Mag **MLOGarithmic** **MLOG**	Single Rectilinear Graph	0	±9.9999E2	10 dB	1E-3 to 1E3	5	4 to 30
Phase **PHASe** **PHAS**	Single Rectilinear Graph	0	±9.9999E2	45 degrees	1E-2 to 1E6	5	4 to 30
Linear Polar Lin/Phase **PLINear** **PLIN**	Polar Graph	5	1E-8 to 9.9999E2	1 U	2E-9 to 1E6	-NA-	-NA-

Table 2-7. Trace Parameters and Coefficients (2 of 4)

Trace Name SCPI Keyword Display Trace Abbreviation	Trace Graph Format	Default Reference Level	Reference Level Range	Default Resolution	Resolution Range Parameter / Division	Default Scale Position	Scale Reference Range - Num of VertIcal Div.
Linear Polar Real/Imag **PLINCOMPlex** **PLIN**	Polar Graph	5	1E-8 to 9.9999E2	1 U	2E-9 to 1E6	-NA-	-NA-
Log Polar Log/Phase **PLOGarithmic** **PLOG**	Polar Graph	0	±9.9999E2	10 dB	1E-5 to 1E6	-NA-	-NA-
Log Polar Real/Imag **PLOGCOMPlex** **PLOGCOMP**	Polar Graph	0	±9.9999E2	10 dB	1E-5 to 1E6	-NA-	-NA-
Power In **PWRIn** **PWRI**	Single Rectilinear Graph	0 dBm	±9.9999E2	10 dB	1E-3 to 1E3	-NA-	-NA-
Power Out **PWROut** **PWRO**	Single Rectilinear Graph	0 dBm	±9.9999E2	10 dB	1E-3 to 1E3	5	4 to 30
Real **REAL** **REAL**	Single Rectilinear Graph	0	±9.9999E2	1 U	1E-5 to 1E6	5	4 to 30
Real and Imaginary **REIMaginary** **REIM**	Double Rectilinear Graphs	TOP = 0 Bottom = 0	±9.9999E2	Top = 1 U Bottom = 1 U	Top = 1E-5 to 1E6 Bottom = 1E-5 to 1E6	Top = 5 Bottom = 5	Top = 4 to 30 Bottom = 40 to 30
Smith (G + jB) Real/Imag **SADCOMPlex** **SADCOMP**	Smith Chart - Admittance (Complex)	-NA-	±9.9999E2	1 U	1E-5 to 1E6	-NA-	-NA-
Smith (G + jB) Lin/Phase **SADLINear** **SADLIN**	Smith Chart - Admittance (Linear)	-NA-	±9.9999E2	10 U	1E-5 to 1E6	-NA-	-NA-

Table 2-7. Trace Parameters and Coefficients (3 of 4)

Trace Name SCPI Keyword Display Trace Abbreviation	Trace Graph Format	Default Reference Level	Reference Level Range	Default Resolution	Resolution Range Parameter / Division	Default Scale Position	Scale Reference Range - Num of Vertical Div.
Smith (G + jB) Log/Phase **SADLOGarithmic** **SADLOG**	Smith Chart - Admittance (Log)	-NA-	±9.9999E2	10 U	1E-5 to 1E6	-NA-	-NA-
Smith (G + jB) Admittance **SADMittance** **SADM**	Smith Chart - Admittance (Admittance)	-NA-	±9.9999E2	10 U	1E-5 to 1E6	-NA-	-NA-
Smith (R + jX) Real/Imag **SCOMPlex** **SCOMP**	Smith Chart - Impedance (Complex)	-NA-	±9.9999E2	10 U	1E-5 to 1E6	-NA-	-NA-
Smith (R + jX) Lin/Phase **SLINear** **SLIN**	Smith Chart - Impedance (Linear)	-NA-	±9.9999E2	10 U	1E-5 to 1E6	-NA-	-NA-
Smith (R + jX) Log/Phase **SLOGarithmic** **SLOG**	Smith Chart - Impedance (Log)	-NA-	±9.9999E2	10 U	1E-5 to 1E6	-NA-	-NA-
Smith (R + jX) Impedance **SMITh** **SMIT**	Smith Chart - Impedance (Impedance)	-NA-	±9.9999E2	10 U	1E-5 to 1E6	-NA-	-NA-
SWR **SWR** **SWR**	Single Rectilinear Graph	0	±9.9999E2	10 U	1E-5 to 1E6	5	4 to 30
Impedance Real & Imaginary **ZCOMPlex** **ZCOMP**	Double Rectilinear Graphs	Top = 0 Ohms Bottom = 0 Ohms	±9.9999E2	Top = 10 Ohms Bottom = 10 Ohms	Top = 1E-5 to 1E6 Bottom = 1E-5 to 1E6	Top = 5 Bottom = 5	Top = 4 to 30 Bottom = 4 to 30

Table 2-7. Trace Parameters and Coefficients (4 of 4)

Trace Name SCPI Keyword Display Trace Abbreviation	Trace Graph Format	Default Reference Level	Reference Level Range	Default Resolution	Resolution Range Parameter / Division	Default Scale Position	Scale Reference Range - Num of VertIcal Div.
Impedance Imaginary **ZIMAGinary** **ZIMAG**	Single Rectilinear Graph	0 Ohms	±9.9999E2	10 Ohms	1E-5 to 1E6	5	4 to 30
Impedance Magnitude **ZMAGNitude** **ZMAGN**	Single Rectilinear Graph	0 Ohms	±9.9999E2	10 Ohms	1E-5 to 1E6	5	4 to 30
Impedance Real **ZREAL** **ZREAL**	Single Rectilinear Graph	0 Ohms	±9.9999E2	10 Ohms	1E-5 to 1E6	5	4 to 30

2-19 Input/Output Data Files

The following is a list of file types that are supported by the VectorStar MS4640B:

Table 2-8. Supported File Types (1 of 3)

File Extension	Description	Command Compatibility
ACD	AutoCal Characterization file.	:MMEM:LOAD <string> | :MMEM:STORe <string>
AHC	All hardware calibration file. On a per system basis, the file contains all hardware calibration data.	:MMEM:LOAD <string> | :MMEM:STORe <string>
AIC	Analog in calibration file. Per-system.	:MMEM:LOAD <string> | :MMEM:STORe <string>
ALC	ALC calibration file. Saves all available ALC calibration for all ports. Per-system.	:MMEM:LOAD <string> | :MMEM:STORe <string>
BMP	Bitmap image file of data display area.	:MMEM:STORe <string> | :MMEM:STORe:IMAGe <string>
CCF	Calibration kit coefficients file.	:MMEMory:LOAD:CKIT
CHA	Setup and Calibration file for all channels.	:MMEM:LOAD <string> | :MMEM:STORe <string>
CHX	Setup and Calibration file for a single channel.	:MMEM:LOAD <string> | :MMEM:STORe <string>
CSV	Comma separated text data file.	:MMEM:STORe <string>
EDL	Embedding/De-embedding array file.	:MMEM:LOAD <string> | :MMEM:STORe <string>
FPC	Flat test port power calibration file	:MMEM:LOAD <string> | :MMEM:LOAD:FLAT <string> :MMEMory:STORe:FLAT{1-2} <string>
INI	Frequency Initialization and Source Initialization table files. Default name is FreqIniTable.ini (for troubleshooting only).	
JPG	JPEG image file of data display area.	:MMEM:STORE <string> :MMEM:STORe:IMAGE <string>
KIT_INFO. “Extension”	The Anritsu Lightning Calibration kit files are supported by the VectorStar MS4640B VNA. The files are usually bundled together on a floppy disk. To use the files, transfer from the floppy disk to a USB memory device. When plugged into an MS4640B Series VNA, the USB drive is identified as drive E:\. Files can be manually loaded by navigating to drive E:\ and selecting the files. The MS4640B VNA Series VNA then loads all of files. Alternatively, use the Lightning LKT command to load all of the calibration kit files. The Lightning calibration kit files all have a base name of “KIT_INFO.Extension” where the extension identifies the connector geometry and gender.	Lightning command: LKT
LMT	Set up for limit lines.	:MMEMory:STORe:LIMit <string> :MMEMory:LOAD:LIMit <string>
LOG	A list of all entries in the VectorStar event log.	:MMEM:STORe <string>

Table 2-8. Supported File Types (2 of 3)

File Extension	Description	Command Compatibility
MFT	Multiple Frequency Table configuration file. Default file name is FreqTable.mft (for troubleshooting).	
PPC	Power sweep power calibration file.	:MMEMory:LOAD:LINearity <string> | :MMEMory:STORe:LINearity{1-2} <string>
PNG	PNG image of data display area.	:MMEM:STORe <string> | :MMEM:STORe:IMAGe <string>
PTC	Pretune calibration file.	
RCVR	Receiver calibration file.	:MMEM:LOAD <string> | :MMEM:STORe <string>
S1P	Data file in S1P format (see S2P below).	:MMEM:STORe <string>
S2P	Data file in S2P standard microwave simulator text format. Includes a controlled header and only one or four S-parameters are saved. If an S2P file is requested, but not all of the S-parameters are currently being measured, a value of 0 (zero) is entered for missing parameters. If a full two-port calibration is applied, all of the S-parameters are measured, even if they do not need to be displayed. The resultant S2P file is complete with all S-parameter information. S2P files can be recalled and displayed as trace memory when they are loaded into the active channel.	:MMEM:STORe <string>
SFT	Single frequency table file (for troubleshooting).	
SGS	Setup file for segmented traces.	:MMEMory:LOAD:FSEGMent <string> | :MMEMory:LOAD:ISEGMent <string>
SLC	Source Local Oscillator (Src LO) calibration file. Per-system.	:MMEM:LOAD <string> | :MMEM:STORe <string>
SQM	Source Quadrupler hardware calibration file	Lightning command: RECALL <string> :MMEM:LOAD <string> | :MMEM:STORe <string>
STA	Setup file for all channels.	:MMEM:LOAD <string> | :MMEM:STORe <string>
STX	Setup file for a single channel.	:MMEM:LOAD <string> | :MMEM:STORe <string>
TDF	Active trace data memory formatted file.	:MMEMory:LOAD:MDATA <string>
TDU	Active trace data memory unformatted file.	:MMEMory:LOAD:MDATA <string>
TMZ	Ten (10) MHz calibration file. Per-system.	:MMEM:LOAD <string> | :MMEM:STORe <string>

Table 2-8. Supported File Types (3 of 3)

File Extension	Description	Command Compatibility
TXT	Active channel trace data text file. Similar to the .csv format described above. A tab-delimited format with an optional descriptive heading in which the data for every trace is saved to a defined location folder. The data for each trace is saved as an X and a Y column to accommodate multiple parameters such as mixed frequency and time domain. Subsequent traces are added as additional columns.The .txt file cannot be recalled into the VNA memory.	:MMEM:STORe <string>

2-20 Status System Reporting

The MS4640B status system consists of the following SCPI-defined status-reporting structures:

- The Instrument Summary Status Byte Group
- The Standard Event Status Group
- The Operation Status Group
- The Questionable Status Group

The following paragraphs describe the registers that make up a status group and explain the status information that each status group provides.

Note	Parallel Polling is not supported in the MS4640B VNA.

Status Group Registers

In general, a status group consists of a condition register, a transition filter, an event register, and an enable register. Each component is briefly described in the following paragraphs.

Condition Register

The condition register is continuously updated to reflect the current status of the MS4640B. There is no latching or buffering for this register, it is updated in real time. Reading the contents of a condition register does not change its contents.

Transition Filter

The transition filter is a special register that specifies which types of bit state changes in the condition register will set corresponding bits in the event register.

- Negative transition filters (NTR) are used to detect condition changes from True (1) to False (0).
- Positive transition filters (PTR) are used to detect condition changes from False (0) to True (1).
- Setting both positive and negative filters True allows an event to be reported anytime the condition changes.
- Transition filters are read-write.
- Transition filters are unaffected by queries or *CLS (clear status) and *RST commands.

Event Register

The event register latches transition events from the condition register as specified by the transition filter. Bits in the event register are latched, and once set they remain set until cleared by a query or a *CLS command Event registers are read only.

Enable Register

The enable register specifies the bits in the event register that can produce a summary bit. The MS4640B logically ANDs corresponding bits in the event and enable registers, and ORs all the resulting bits to obtain a summary bit. Summary bits are recorded in the Summary Status Byte. Enable registers are read-write. Querying an enable register does not affect it.

Status Group Reporting

The state of certain MS4640B hardware and operational events and conditions can be determined by programming the status system. Three lower status groups provide status information to the Summary Status Byte group. The Summary Status Byte group is used to determine the general nature of an event or condition and the other status groups are used to determine the specific nature of the event or condition. The following paragraphs explain the information that is provided by each status group. Programming commands for the status system, including examples of command usage, can be found in Chapter 5, “SCPI Commands”.

Summary Status Byte Group

The Summary Status Byte group, consisting of the Summary Status Byte Enable register and the Summary Status Byte, is used to determine the general nature of a MS4640B event or condition. The bits in the Summary Status Byte provide the following:

Table 2-9. Status Byte Group

Bit #	Bit Name	Description
0,1	Not Used	These bits are always set to 0.
2	Error Queue (ERRQ)	Set to indicate the Error Queue contains data. The Error Query command can then be used to read the error message(s) from the queue.
3	Questionable Event (QUEST)	Set to indicate the Questionable Status summary bit has been set. The Questionable Status Event register can then be read to determine the specific condition that caused the bit to be set.
4	Message Available (MAV)	Set to indicate that the MS4640B has data ready in its output queue.
5	Standard Event (STD)	Set to indicate that the Standard Event Status summary bit has been set. The Standard Event Status register can then be read to determine the specific event that caused the bit to be set.
6	Master Summary Status (MSS/RQS)	Set to indicate that the MS4640B has at least one reason to require service. This bit is also called the Master Summary Status Bit (MSS). The individual bits in the Status Byte are ANDed with their corresponding Service Request Enable Register bits, then each bit value is ORed and input to this bit.
7	Operation Event (OPER)	Set to indicate that the Operation Status summary bit has been set. The Operation Status Event register can then be read to determine the specific condition that caused the bit to be set.

Standard Event Status Group

The Standard Event Status group, consisting of the Standard Event Status register (an Event register) and the Standard Event Status Enable register, is used to determine the specific event that set bit 5 of the Summary Status Byte. The bits in the Standard Event Status register provide the following:

Table 2-10. Standard Event Status Group

Bit #	Bit Name	Description
0	Operation Complete (OP)	Set to indicate that all pending MS4640B operations were completed following execution of the "*OPC" command. For more information, see the descriptions of the *OPC, *OPC?, and *WAI commands in Chapter 3, "IEEE Commands".
1	Not Used	The bit is always set to 0.
2	Query Error	Set to indicate that a query error has occurred.
3	Device Dependent Error	Set to indicate that a device-dependent error has occurred.
4	Execution Error	Set to indicate that an execution error has occurred.
5	Command Error	Set to indicate that a command error (usually a syntax error) has occurred.
6	Not Used	This bit should be set to 0 (zero).
7	Power ON	Set to indicate that the MS4640B is powered ON and in operation.

Operation Status Group

The Operation Status group, consisting of the Operation Condition register, the Operation Positive Transition register, the Operation Negative Transition register, the Operation Event register, and the Operation Event Enable register, is used to determine the specific condition that set bit 7 in the Summary Status Byte. The bits in the Operation Event register provide the following:

Table 2-11. Operation Status Group

Bit #	Bit Name	Description
0	Calibration Complete	Set to indicate that a calibration is complete.
1	Sweep Complete	Set to indicate that a sweep is complete. Note that the Sweep Complete Bit will not be set unless the sweep was started by an appropriate trigger commands. For examples of use, see the “TRS” command in the Lightning 37xxxx Command chapter in the Programming Manual Supplement. Also see “:TRIGger[:SEQuence] Subsystem” on page 5-552 in Chapter 5, “SCPI Commands”.
2-3	Not Used	These bits should be set to 0 (zero).
4	Waiting for Trigger	Set to indicate that the MS4640B is in an armed “wait for trigger” state.
6-15	Not Used	These bits should be set to 0 (zero).

Questionable Status Register

The Questionable Status Register consists of the Questionable Condition register, the Questionable Positive Transition register, the Questionable Negative Transition register, the Questionable Event register, and the Questionable Event Enable register.

The Questionable Status Register is used to determine the specific condition that set bit 3 in the Summary Status Byte. The bits in the Questionable Event register provide the following:

Table 2-12. Questionable Status Register

Bit #	Bit Name	Description
0	New Service Log Entry	Set to indicate that a new entry has been made to the Windows service log.
1	Limit Failure	Set to indicate that trace data is outside a limit line boundary.
2	RF Unleveled	Set to indicate that an RF unleveled condition exists.
3	Unlocked	Set to indicate that an internal PLL unlocked condition exists.
4-15	Not Used	These bits should be set to 0 (zero).

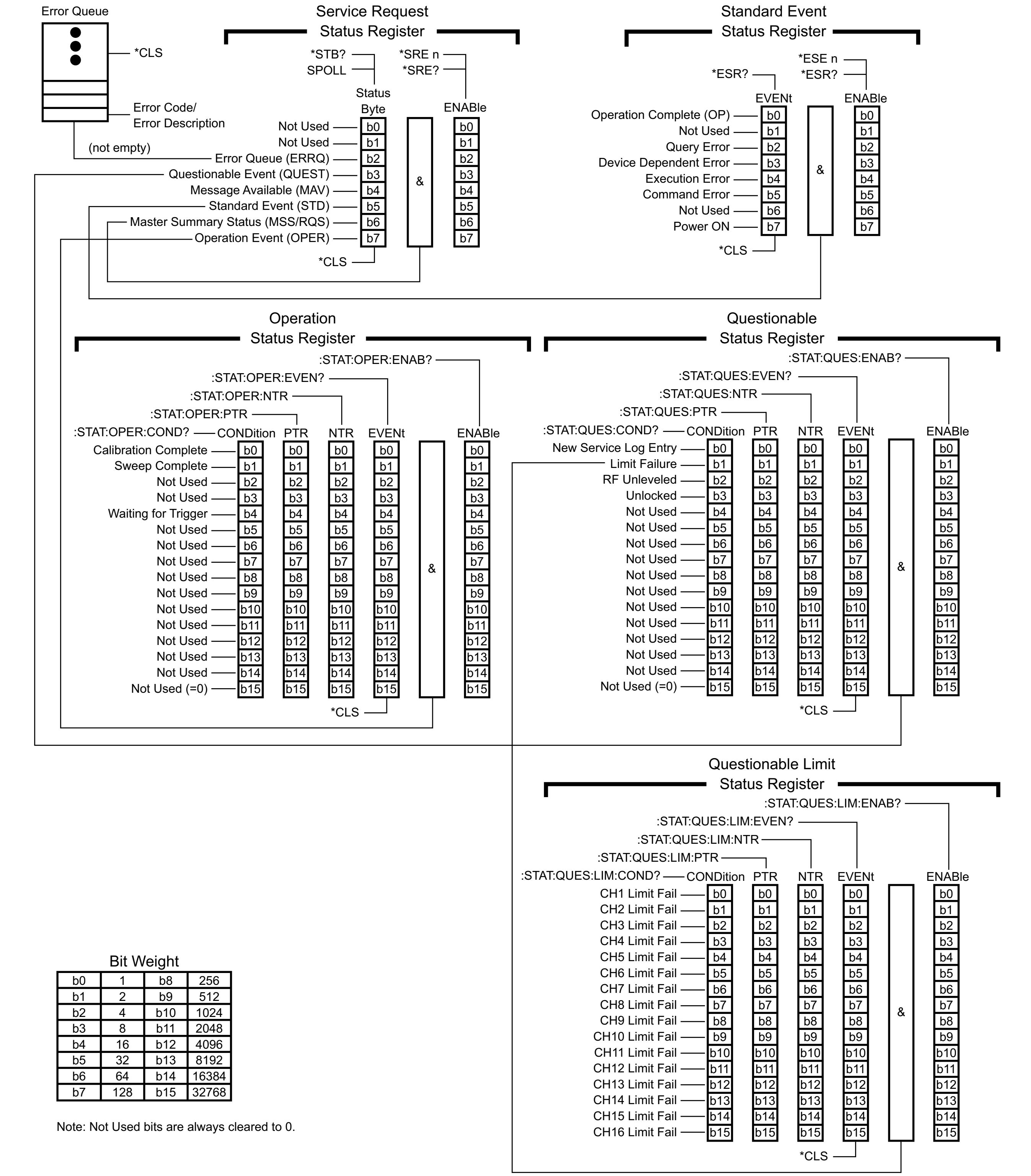
Questionable Limits Status Register

The Questionable Limits Status Register (QLSR) consists of the Questionable Limits Condition register, the Questionable Limits Event register, the Positive and Negative Transition Filters, and the Questionable Limits Event Enable register.

The QLSR is used to determine the channels that continuous limits testing failures and set Bit B1 of the Questionable Status Register. The bits in the QLSR provide the information described in the table below.

Table 2-13. Questionable Limits Status Register (QLSR)

Bit #	Bit Name	Description
0	Channel1Fail	Limits testing on Channel 1 detected a failure
1	Channel2Fail	Limits testing on Channel 2 detected a failure
2	Channel3Fail	Limits testing on Channel 3 detected a failure
3	Channel4Fail	Limits testing on Channel 4 detected a failure
4	Channel5Fail	Limits testing on Channel 5 detected a failure
5	Channel6Fail	Limits testing on Channel 6 detected a failure
6	Channel7Fail	Limits testing on Channel 7 detected a failure
7	Channel8Fail	Limits testing on Channel 8 detected a failure
8	Channel9Fail	Limits testing on Channel 9 detected a failure
9	Channel10Fail	Limits testing on Channel 10 detected a failure
10	Channel11Fail	Limits testing on Channel 11 detected a failure
11	Channel12Fail	Limits testing on Channel 12 detected a failure
12	Channel13Fail	Limits testing on Channel 13 detected a failure
13	Channel14Fail	Limits testing on Channel 14 detected a failure
14	Channel15Fail	Limits testing on Channel 15 detected a failure
15	Channel16Fail	Limits testing on Channel 16 detected a failure



Bit Weight

b0	1	b8	256
b1	2	b9	512
b2	4	b10	1024
b3	8	b11	2048
b4	16	b12	4096
b5	32	b13	8192
b6	64	b14	16384
b7	128	b15	32768

Note: Not Used bits are always cleared to 0.

Figure 2-3. Status Register Structure

2-21 Trigger System

The MS4640B trigger system is used to synchronize analyzer actions with software trigger commands. The VNA follows the layered trigger model used in SCPI instruments. The following paragraphs describe the operation of the analyzer's trigger system. A sample logic flowchart of the trigger model is shown in Figure 2-4. A sample timing reference of the trigger model is shown in Figure 2-5.

Trigger Modes

The trigger system supports four different trigger modes:

- **Internal Trigger Mode**

  This is an automatic triggered point-by-point measurement that is internally controlled by the DSP software.

- **Manual Trigger Mode**

  Manual mode is triggered by the user from the front panel to start a measurement based on a per point, a per sweep (or a per port), or a per channel (or all channels) trigger mode.

- **GPIB Trigger Mode**

  GPIB mode is triggered by a GPIB trigger command to start a measurement based on a point-per-point, a sweep-per-sweep (or a port-per-port), or a channel-per-channel (or all channels).

- **External Trigger Mode**

- External mode is triggered through the rear panel input of the instrument to start a measurement based on a point-per-point, a sweep-per-sweep (or a port-per-port), or a channel-per-channel (or all channels) trigger mode.
  - The external trigger system allows the user to select a positive or negative edge trigger to start the measurement.
  - A trigger delay can also be applied to a measurement right after an external trigger is received by the instrument and just before the measurement begins.
  - The external trigger has an additional feature to handle trigger handshaking, which uses the "Ready for Trigger" and "Trigger Output" output signals through the rear panel of the instrument.
  - The Ready for Trigger signal is sent from the instrument to the rear panel output when the system is ready to accept an external trigger.
  - The Trigger Output pulse signal is sent to the rear panel when the system has completed a measurement.

The following diagram is a flowchart of the triggering logic:

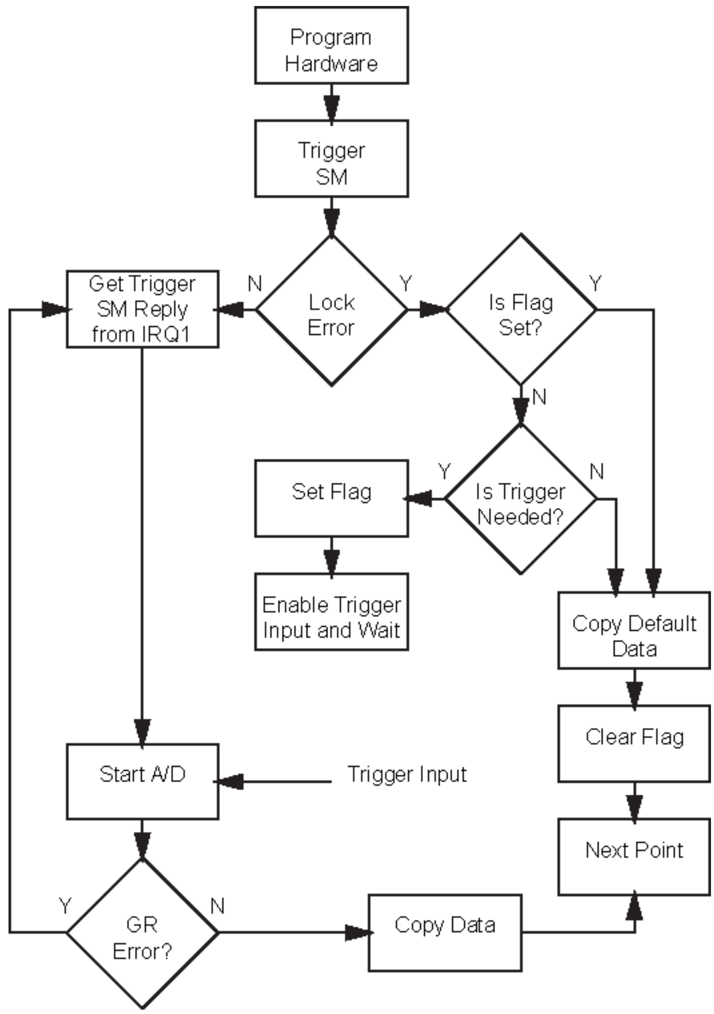


Figure 2-4. Triggering Logic Flowchart

Trigger Timing

The timing diagram below illustrates the general measurement sequence of per-point triggering using a positive trigger edge. Trigger Delay, Ready for Trigger, and Trigger Output are only used by an external trigger.

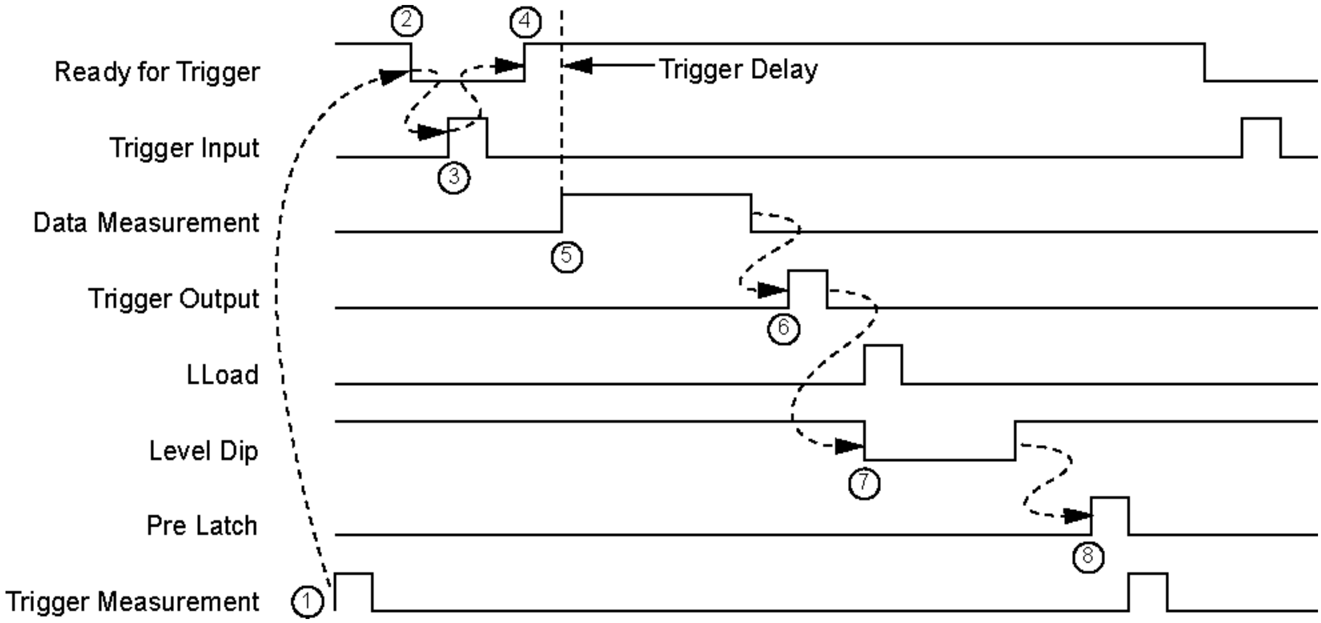


Figure 2-5. Sample Trigger Model

The trigger sequence with trigger handshake turned on is as follows (Trigger Handshake enables “Ready for Trigger” and “Trigger Output”:

1. A trigger measurement is received from the State machine.
2. Ready for Trigger (rear panel BNC output) is set to low to indicate that the instrument is ready for trigger.
3. External trigger (rear panel BNC input) is received.
4. After the external trigger is received, the Ready for Trigger (rear panel BNC output) is set to high to indicate that the system is not ready for trigger and the trigger delay is added.
5. The data measurement is started.
6. At the completion of the data measurement, the Trigger Output (rear panel BNC output) is pulsed to indicate that the measurement is completed.
7. The Level Dip and/or the LLoad pulse are executed.
8. The next frequency is preloaded and the State machine is triggered.

Note	Trigger handshaking is enabled with: `:TRIGger[:SEQuence]:EXTernal:HANDshake[:STATe]` If handshaking is not in use, steps 2, 4, and 6 are deleted, but the trigger delay is still present.

2-22 Calibration Component Parameters

The calibration component parameter values depend on the calibration kit used and the reset status of the instrument. The tables below, starting with Table 2-14, “Loads and Through-Line Values” on page 2-38 and ending with Table 2-21 on page 2-43 summarize parameters related to calibration components and list the factory default values for the various connector coefficients and lengths. These values may change if calibration kits are loaded that overwrite them.

A Factory Default using the :SYStem:PRESet:ZERo command restores all connector values to those in the tables below. After the restoration, a Factory Default also performs a Default Default. A Default Default (System Default not set to USER) changes the connector type to the appropriate connector type based on the model number.

The following general calibration component parameters should also be noted:

- A Default Default (System Default not set to USER) will change the Microstrip Kit to 10 Mil.
- A Default Default (System Default not set to USER) will change the Waveguide Kit to WR10.

Loads and Through Lines

The standard values for Loads and Through Lines are listed in the table below.

Table 2-14. Loads and Through-Line Values

Type	Parameter	Value	Units
Loads	Impedance	50	Ohms
	Resistance	50	Ohms
Through Lines	Impedance	50	Ohms
	Length	0	Meters
	Loss	0	dB
	Frequency	0	Hz

Other Connector Coefficients

The default connector coefficients apply if not overwritten by the connector values loaded from a calibration kit.

Table 2-15. Default Connector Coefficients (1 of 2)

Value	Type (gender)	Type (gender)	Type (gender)	Type (gender)	Type (gender)	Type (gender)
Type (gender)	**SMA (male)**	**SMA (female)**	**K (male)**	**K (female)**	**N (male)**	**N (female)**
OpenC0	23E–15	26E–15	–1.5E–15	–1.0E–15	125.0E–15	65.0E–15
OpenC1	–550E–27	–550E–27	720E–27	650E–27	0.0E–27	0.0E–27
OpenC2	10.0E–36	10.0E–36	–23.0E–36	–23.0E–36	0.0E–36	0.0E–36
OpenC3	0.7E–45	0.5E–45	0.35E–45	0.35E–45	10.0E–45	6.0E–45
OpenOffsetLength	5.0E–3	5.0E–3	5.0E–3	5.0E–3	M DUT = 0.0E–3	F Test Port = 20.3E–3
					M Test Port = 8.97E–3	F DUT = 0.0E–1
ShortL0	0	0	0	0	0	0
ShortL1	0	0	0	0	0	0
ShortL2	0	0	0	0	0	0
ShortL3	0	0	0	0	0	0
ShortOffsetLength	5.0E–3	5.0E–3	5.0E–3	5.0E–3	20.37E–3	8.97E–3
Type (gender)	**GPC3.5 (male)**	**GPC3.5 (female)**	**GPC7 (none)**	**V (male)**	**V (female)**	**TNC (male)**
OpenC0	24E–15	24E–15	91E–15	–1.0E–15	0.0E–15	79.0E–15
OpenC1	–425.0E–27	–250.0E–27	–220.0E–27	–275.0E–27	–200.0E–27	0.0E–27
OpenC2	10.0E–36	10.0E–36	75.0E–36	6.25E–36	5.0E–36	40.0E–36
OpenC3	0.6E–45	0.5E–45	1.3E–45	0.0E–45	0.0E–45	0.0E–45
OpenOffsetLength	5.0E–3	5.0E–3	0.0E–3	4.75E–3	4.75E–3	18.05E–3
ShortL0	0	0	0	0	0	0
ShortL1	0	0	0	0	0	0
ShortL2	0	0	0	0	0	0
ShortL3	0	0	0	0	0	0
ShortOffsetLength	5.0E–3	5.0E–3	0.0E–3	5.10E–3	5.10E–3	18.05E–3

Table 2-15. Default Connector Coefficients (2 of 2)

Value	Type (gender)	Type (gender)	Type (gender)	Type (gender)	Type (gender)	Type (gender)
Type (gender)	**TNC (female)**	**N75 (male)**	**N75 (female)**	**7/16 (male)**	**7/16 (female)**	–
OpenC0	79E–15	55.0E–15	23.0E–15	32.0E–15	32.0E–15	–
OpenC1	0.0E–27	0.0E–27	0.0E–27	100.0E–27	100.0E–27	–
OpenC2	40.0E–36	0.0E–36	0.0E–36	–50.0E–36	–50.0E–36	–
OpenC3	0.0E–45	0.0E–45	0.0E–45	100.0E–45	100.0E–45	–
OpenOffsetLength	25.27E–3	0.0E–3	0.0E–3	20.0E–3	20.0E–3	–
ShortL0	0	0	0	0	0	–
ShortL1	0	0	0	0	0	–
ShortL2	0	0	0	0	0	–
ShortL3	0	0	0	0	0	–
ShortOffsetLength	25.27E–3	0.0E–3	0.0E–3	20.0E–3	20.0E–3	–

W1 Calibration Kit

A W1 Calibration Kit is slightly different because it has to cover such a wide frequency range.

It has a selection of components to use in a SOLT Calibration that covers the 70 kHz to 70 GHz range. A Male and Female Open and a Male and Female Short with a set of coefficients characterize them over that range.

It has a selection of components to use in an SSST Calibration which covers the 70 kHz to 70 GHz range. Three Male and Female Shorts with a set of coefficients characterize them over that range.

Physically, there are only 3 shorts: ShortX, ShortY, and ShortZ. ShortX has two different sets of coefficients (the offset lengths are the same). The first set of coefficients are good over the 70 kHz to 70 GHz range and the second set of coefficients (referred to as OffsetShort) are good over the 70 GHz to 110 GHz range.

Table 2-16. W1 Calibration Kit Connector Parameters (1 of 2)

Geometry	W1 Male	W1 Female
OpenC0	9.63E–15	4.43E–15
OpenC1	374.73E–27	–1.109E–27
OpenC2	–4.96E–36	14.65E–36
OpenC3	0.0154E–45	–0.1768E–45
OpenOffsetLength	1.510E–3	1.930E–3
ShortL0	0.98E–12	0.98E–12
ShortL1	2.48E–24	2.48E–24
ShortL2	–1.02E–33	–1.05E–33
ShortL3	0.0164E–42	0.0164E–42
ShortOffsetLength	2.020E–3	2.020E–3

Table 2-16. W1 Calibration Kit Connector Parameters (2 of 2)

Geometry	W1 Male	W1 Female
OffsetShort1L0	–4.1930E–12	3.13E–12
OffsetShort1L1	1308.5E–24	–971.86E–24
OffsetShort1L2	–25.1370E–33	25.010E–33
OffsetShort1L3	0.1282E–42	–0.1464E–42
OffsetShort1OffsetLength	2.020E–3	2.020E–3
OffsetShort2L0	0.9450E–12	–2.13E–12
OffsetShort2L1	544.52E–24	1117.4E–24
OffsetShort2L2	–12.32E–33	–20.4450E–33
OffsetShort2L3	0.0724E–42	0.0968E–42
OffsetShort2OffsetLength	2.650E–3	2.650E–3
OffsetShort3L0	3.0800E–12	–0.0018E–12
OffsetShort3L1	–106.12E–24	–150.2460E–24
OffsetShort3L2	3.50E–33	3.1380E–33
OffsetShort3L3	–0.0218E–42	–0.0107E–42
OffsetShort3OffsetLength	3.180E–3	3.180E–3

Sliding Load Cutoff Frequency

The general sliding load frequency parameters are:

- The V, 2.4mm and W1 connector sliding load cutoff frequency is 4.0E9 Hz.
- Any other connector sliding load cutoff frequency is 2.0E9 Hz.

Table 2-17. Sliding Load Cutoff Frequency

Connector	Sliding Load Cutoff Frequency	–	–
V	4.0E9 Hz	–	–
2.4mm	4.0E9 Hz	–	–
W1	4.0E9 Hz	–	–
Other Connectors	2.0E9 Hz	–	–

Microstrip Kit Common Values

The following microstrip kit values are set.

Table 2-18. MIcrostrip Calibration Kit Common Values

Parameter	Value	–	–
Microstrip Default Impedance	50 Ohms	–	–
Microstrip Substrate Dielectric	9.96	–	–
Microstrip Effective Dielectric	6.69	–	–

Table 2-19. Microstrip Kit Dimensions

Parameter	10 Mil	15 Mil	25 Mil
Microstrip Thickness	0.254E–3	0.381E–3	0.635E–3
Microstrip Width	0.23876E–3	0.35814E–3	0.5969E–3

Table 2-20. Waveguide Frequencies and Lengths

Parameter	WR10	WR12	WR15
Cutoff Frequency	59.0143E9	48.3723E9	39.8766E9
SHORT1 Offset Length	2.537E–3	2.654E–03	2.793E–03
SHORT2 Offset Length	3.612E–3	3.963E–3	4.380E–3
SHORT3 Offset Length	0	0	0

* The WR10, WR12 and WR15 values are valid for the SSLT calibration method only. All values are zero for the SSST calibration method.

Table 2-21. SOLX, SSLT, Microstrip, and Waveguide Kit Names and Connectors

Kit Name	Defined Connectors	User-Defined Connectors
SOLX Kit	2.4 mm GPC-3.5 K-Conn N-Conn SMA TNC V-Conn W1-Conn 7/16 GPC-7 N-Conn(75)	User-Defined1 User-Defined2 User-Defined3 User-Defined4 User-Defined5 User-Defined6 User-Defined7 User-Defined8
SSLT Kit	W1-Conn	User-Defined1 User-Defined2 User-Defined3 User-Defined4 User-Defined5 User-Defined6 User-Defined7 User-Defined8
Microstrip Kit	10-Mil-Kit 15-Mil-Kit 25-Mil-Kit	User-Defined1 User-Defined2 User-Defined3 User-Defined4 User-Defined5 User-Defined6 User-Defined7 User-Defined8
Waveguide Kit	WR10 WR12 WR15	User-Defined1 User-Defined2 User-Defined3 User-Defined4 User-Defined5 User-Defined6 User-Defined7 User-Defined8

2-23 Notes on Calibration Commands

This section provides an overview of the calibration commands and when they should be used.

Setting Up a Two-Port Calibration

The commands listed in this section work on the first two ports of the instrument on either 2-port or 4-port configurations. The FULL1 calibration is set up with the following calibration commands:

1. Calibration method

:SENSe{1-16}:CORRection:COLLect:METHod

Available calibration methods: AUTOcal | LRL | LRM | SOLR | SOLT | SSLT | SSST

2. Calibration type

:SENSe{1-16}:CORRection:COLLect:1P2PF

:SENSe{1-16}:CORRection:COLLect:1P2PR

:SENSe{1-16}:CORRection:COLLect:FULL1

:SENSe{1-16}:CORRection:COLLect:FULL2

:SENSe{1-16}:CORRection:COLLect:FULLB

:SENSe{1-16}:CORRection:COLLect:RESP1

:SENSe{1-16}:CORRection:COLLect:RESPB

:SENSe{1-16}:CORRection:COLLect:TFRB

:SENSe{1-16}:CORRection:COLLect:TFRF

:SENSe{1-16}:CORRection:COLLect:TFRR

:SENSe{1-16}:CORRection:COLLect:TYPe?

3. Line type

:SENSe{1-16}:CORRection:COLLect:LINE

Available line types: COAXial | MICROstrip | NONDISpersive | WAVEguide

4. Load type

:SENSe{1-16}:CORRection:COLLect:LOAD

Available load types: FIXED | SLIDING

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:SELect

Available loads: LOAD1 | LOAD2

5. Calibration port

:SENSe{1-16}:CORRection:COLLect:PORT

Available calibration ports: PORT1 | PORT2 | PORTP12

Setting Up a Four-Port Calibration

4-port calibrations are more complex and are hence divided into six broad categories:

- "Reflection Response Calibration"
- "Full One-Port Calibration"
- "Transmission Response Calibration"
- "Full Two-Port Calibration"
- "Full Three-Port Calibration"
- "Full Four-Port Calibration"

Reflection Response Calibration

Under the reflection response calibration type, one can specify up to four individual 1-port response calibrations to be performed using the following command:

```
:SENSe{1-16}:CORRection:COLLect:PORT {1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34
| 123 | 124 | 134 | 234 | 1234}:RESP1
```

For instance, to perform a reflection response calibration on ports 2, 3, and 4, the command is:

```
:SENS1:CORR:COLL:PORT234:RESP1
```

The following command is used to define the components used in the reflection calibration:

```
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:REFLection:COMPonent
```

Available reflection components are: NONE | OPEN | SHORt | OFSH1 | OFSH2 | OFSH3

Full One-Port Calibration

With a full 1-port calibration type, one can specify up to four individual 1-port calibrations to be performed using the following command:

```
:SENSe{1-16}:CORRection:COLLect:PORT {1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34
| 123 | 124 | 134 | 234 | 1234}:FULL1
```

For instance, to perform a full 1-port calibration on ports 2, 3, and 4, the command is:

```
:SENS1:CORR:COLL:PORT234:FULL1
```

Transmission Response Calibration

Under the transmission response calibration type, one can specify up to six, 2-port combinations:

```
12 | 13 | 14 | 23 | 24 | 34
```

using three response methods that are stored in a list:

```
TFRF: transmission frequency response, forward
TFRR: transmission frequency response, reverse
TFRB: transmission frequency response, both.
```

First, clear the list using:

```
:SENSe{1-16}:CORRection:COLLect:TFR:CLEar
```

Next, use a combination of the following three commands:

```
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRB
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRF
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRR
```

For instance, to specify a forward response calibration on port12, a reverse response calibration on port13, and both forward and reverse calibrations on port14, the commands are:

```
:SENS1:CORR:COLL:TFR:CLE
:SENS1:CORR:COLL:PORT12:TFRF
:SENS1:CORR:COLL:PORT13:TFRR
:SENS1:CORR:COLL:PORT14:TFRB
```

Full Two-Port Calibration

With a full 2-port calibration type, one can specify two, 2-port calibrations to be performed with six, 2-port combinations:

```
12 | 13 | 14 | 23 | 24 | 34
```

The port pairs selected must be port exclusive. For instance, if the first calibration is on port23, then the second, if specified, must be on port14. There are 3 calibration types that can be specified:

```
FULL2 | 1P2PF | 1P2PR
```

This is accomplished using one of the commands below for the first calibration:

```
:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PF
:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PR
:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2
```

And one of the following commands below for the second calibration:

```
:SENSe{1-16}:CORRection:COLLect:CALB:1P2PF
:SENSe{1-16}:CORRection:COLLect:CALB:1P2PR
:SENSe{1-16}:CORRection:COLLect:CALB:FULL2
```

For instance, the commands below specify a full, 2-port calibration on port12 and use a one path, 2-port forward calibration as the second calibration:

```
:SENS1:CORR:COLL:PORT12:FULL2
:SENS1:CORR:COLL:CALB:1P2PF
```

Full Three-Port Calibration

With a full 3-port calibration type, one can specify a 3-port calibration to be performed on the indicated ports with the following command:

```
:SENSe{1-16}:CORRection:COLLect:PORT{123 | 124 | 134 | 234}:FULL3
```

In addition to selecting the ports above, two or three thru lines must be measured using the commands below:

```
:SENSe{1-16}:CORRection:COLLect:THRu:CLEar
:SENSe{1-16}:CORRection:COLLect:THRu:ADD
```

The arguments of the above command are:

```
THRu12 | THRu13 | THRu14 | THRu23 | THRu24 | THRu34
```

For example, to perform a full, 3-port calibration on ports 1, 3, and 4 with thru lines on port pairs 13, 14, and 34, send the following commands:

```
:SENS1:CORR:COLL:PORT134:FULL3
:SENS1:CORR:COLL:THR:CLE
:SENS1:CORR:COLL:THR:ADD THR13
:SENS1:CORR:COLL:THR:ADD THR14
:SENS1:CORR:COLL:THR:ADD THR34
```

Full Four-Port Calibration

With a full 4-port calibration type, one can specify a 4-port calibration to be performed on all four ports with the following command:

```
:SENSe{1-16}:CORRection:COLLect:FULL4
```

In addition, three to six thru lines must be measured using the commands below:

```
:SENSe{1-16}:CORRection:COLLect:THRu:CLEar
:SENSe{1-16}:CORRection:COLLect:THRu:ADD
```

Available throughs are:

```
THRu12 | THRu13 | THRu14 | THRu23 | THRu24 | THRu34
```

For example, to perform a full, 4-port calibration with thru lines on port pairs 12, 13, 14, and 24, send the following commands:

```
:SENS1:CORR:COLL:FULL4
:SENS1:CORR:COLL:THR:CLE
:SENS1:CORR:COLL:THR:ADD THR12
:SENS1:CORR:COLL:THR:ADD THR13
:SENS1:CORR:COLL:THR:ADD THR14
:SENS1:CORR:COLL:THR:ADD THR24
```

Defining the Calibration Standards

The following command sets the connector type:

```
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:CONNector
```

The connector types are as follows (the second letter is the gender: F for female, M for male, N for no gender):

Table 2-22. Connector Type Abbreviations and Descriptions

Connector Type	Connector Description
CF2, CM2	2.4 mm Female, 2.4 mm Male
CF3, CM3	3.5 mm
CFK, CMK	K
CFN, CMN	N
CFS, CMS	SMA
CFC, CMC	TNC (predefined types)
CFV, CMV	V
CF1, CM1	W1
CF716, CM716	7/16
CFN75, CMN75	N (75 ohm)
CF-2V, CM-2V	Anritsu 2.4 mm Female, 2.4 mm Male
CNG	GPC7 genderless
CFU1, CMU1 CFU2, CMU2 CFU3, CMU3 CFU4, CMU4 CFU5, CMU5 CFU6, CMU6 CFU7, CMU7 CFU8, CMU8	User Defined 1 User Defined 2 User Defined 3 User Defined 4 User Defined 5 User Defined 6 User Defined 7 User Defined 8

Use the following command to load a calibration kit file with its path and name as string data:

`:MMEMory:LOAD:CKIT`

The many calibration standard types are divided into four categories of OPEN, SHORT, LOAD, and THRU (or THROUGH) as described in the following sections.

OPEN

An OPEN standard has the following parameters that define its electrical behavior:

- C0, C1, C2 and C3 are power series coefficients used to calculate capacitance as follows:

  `C = C0 + C1*f + C2*f^2 + C3*f^3`

These coefficients are often displayed in scientific notation as shown below:

```
C0 = number x 10E-15
C1 = number x 10E-27
C2 = number x 10E-36
C3 = number x 10E-45
```

If one enters a number for Cx whose magnitude is > 10E-5, then it is assumed that the number must be multiplied by the appropriate power of 10 shown above to determine the coefficient. Otherwise, the coefficient value is taken as is.

- OFFSET is the offset length of the load expressed in meters

Note	The parameters of the predefined types cannot be changed. Only the parameters of the User Defined types can be changed.

The following commands are used to change the OPEN standard parameters:

```
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:OFFS
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:SERIAL
```

SHORT

A SHORT standard has the following parameters that define its electrical behavior:

L0, L1, L2 and L3 are power series coefficients used to calculate inductance as follows:

`L = L0 + L1*f + L2*f^2 + L3*f^3`

These coefficients are often displayed in scientific notation as shown below:

```
L0 = number x 10E-12
L1 = number x 10E-24
L2 = number x 10E-33
L3 = number x 10E-42
```

If one enters a number for Lx whose magnitude is > 10E-5, then it is assumed that the number must be multiplied by the appropriate power of 10 shown above to determine the coefficient. Otherwise, the coefficient value is taken as is.

- OFFSET is the offset length of the load expressed in meters

There are actually four SHORTS that one may encounter; however, one may be working with more than one at a time so numerics are used to differentiate them as follows:

```
SHORT
SHORT1
SHORT2
SHORT3
```

The following commands are used to change the SHORT standard parameters:

```
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:OFFS
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:SERIAL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:OFFS
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:SERIAL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:OFFS
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:SERIAL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:OFFS
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:SERIAL
```

LOAD

A LOAD standard has the following parameters that define its electrical behavior:

- C0 is a capacitance term
- LO, L1, L2, L3 are power series coefficients used to calculate inductance as follows:

  L = L0 + L1*f + L2*f^2 + L3*f^3
- R is the resistance of the load
- Z0 is the characteristic impedance
- OFFSET is the offset length of the load expressed in meters

Most calibration kits have two loads; therefore, they are differentiated by naming them LOAD1 and LOAD2. Use the following commands to modify the LOAD parameters:

```
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFFS
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:R
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:Z0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:SERIAL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFFS
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:R
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:Z0
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:SERIAL
```

THRU (or THROUGH)

A THRU (technically a “through”) standard has the following parameters that define its electrical behavior:

- LENGTH is the length of the line
- LOSS is the loss of the line
- FREQUENCY is the frequency at which the loss was measured
- Z0 is the Characteristic impedance
- USERECIPROCAL is not actually an electrical parameter. It is merely a flag to notify the calibrator that it should use a reciprocal type of calculation.

Use the following commands to modify the THRU parameters:

```
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU:FREQuency
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU:LENGth
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU:LOSS
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU:Z0
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU:RECIProcal
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU:LABEL
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU:SERIAL
```

Performing the Calibration

Once all of the calibration setup parameters described above have been complete, the actual measurements for the calibration type and methods can be performed. Each calibration type and method requires measuring the appropriate standards using the commands listed below:

```
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD1
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD2
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD3
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD4
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD5
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD6
```

2-Port Isolation and Thru:

```
:SENSe{1-16}:CORRection:COLLect:ISOL
:SENSe{1-16}:CORRection:COLLect:THRU
```

4-Port Isolation and Thru:

```
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:ISOL
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRU
```

2-Port and 4-Port Thru Update:

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:UPDate

Once the measurements are complete, the correction coefficients must be calculated and the calibration corrections must be applied (turned on):

:SENSe{1-16}:CORRection:COLLect:SAVE

:SENSe{1-16}:CORRection:ISOLation:STATe

:SENSe{1-16}:CORRection:STATe

To simulate a calibration, use one of the commands below to specify the calibration type:

:SENSe{1-16}:CORRection:COEFFicient:1P2PF

:SENSe{1-16}:CORRection:COEFFicient:1P2PR

:SENSe{1-16}:CORRection:COEFFicient:FULL1

:SENSe{1-16}:CORRection:COEFFicient:FULL2

:SENSe{1-16}:CORRection:COEFFicient:FULLB

:SENSe{1-16}:CORRection:COEFFicient:RESP1

:SENSe{1-16}:CORRection:COEFFicient:RESPB

:SENSe{1-16}:CORRection:COEFFicient:TFRB

:SENSe{1-16}:CORRection:COEFFicient:TFRF

:SENSe{1-16}:CORRection:COEFFicient:TFRR

The commands below are for simulations with a 4-port test set attached to the VNA:

:SENSe{1-16}:CORRection:COEFficient:PORT{1-4}:1P2PF

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PR

:SENSe{1-16}:CORRection:COEFficient:PORT{1-4}:FULL1

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULLB

:SENSe{1-16}:CORRection:COEFficient:PORT{1-4}:RESP1

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:RESPB

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRB

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRF

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRR

:SENSe{1-16}:CORRection:COEFficient:PORT{PORT,123 | 124 | 134 | 234}:FULL3

:SENSe{1-16}:CORRection:COEFficient:FULL4

The following command inputs and outputs the specified correction coefficients:

:SENSe{1-16}:CORRection:COEFficient

The correction coefficients must be specified using one of the character data arguments below:

- 2-Port:

  ED1 | EP1S | ET11 | ET21 | EP2L | EX21 | ED2 | EP2S | ET22 | ET12 | EP1L | EX12
- 4-Port only:

  ED3 | ET31 | ET32 | ET13 | ET23 | ET33 | EP3L | EP3S | EX31 | EX32 | EX13 | EX23 | ED4 | ET14 | ET41 | ET24 | ET42 | ET34 | ET43 | ET44 | EP4L | EP4S | EX14 | EX24 | EX34 | EX41 | EX42 | EX43

AutoCal

The AutoCal calibration method (calibration using a compatible Precision Automatic Calibrator module) must first be specified using the command below:

```
:SENSe{1-16}:CORRection:COLLect AUTO
```

Then the calibration type is selected using the commands described above in “Setting Up a Two-Port Calibration” on page 2-44 or “Setting Up a Four-Port Calibration” on page 2-45.

Without a 4-port test set, the calibration types supported are FULL1 and FULL2. With a 4-port test set, the calibration types supported are FULL1, FULL2, and FULL4.

Note	Multiple calibrations are not supported with AutoCal.

With the exception of the FULL4 calibration, use the commands described above to set the desired calibration type. For a FULL4 calibration, use the calibration type command:

```
:SENSe{1-16}:CORRection:COLLect:ECAL:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL4
```

The FULL4 calibration is achieved by performing two FULL2 calibrations on independent port pairs. One to four thru lines must also be measured using the commands listed above in “Full Three-Port Calibration” on page 2-46.

The following command sets the autocal box orientation manually:

```
:SENSe{1-16}:CORRection:COLLect:ECAL:ORIentation
```

This command inputs a list of up to four comma-separated items:

```
L1 | L2 | L3 | L4 | R1 | R2 | R3 | R4 | L1R2 | L1R3 | L1R4 | L2R3 | L2R4 | L3R4 |
R1L2 | R1L3 | R1L4 | R2L3 | R2L4 | R3L4 | R2L1 | R3L1 | R4L1 | R3L2 | R4L2 | R4L3
| L2R1 | L3R1 | L4R1 | L3R2 | L4R2 | L4R3
```

The following command specifies substituting a true thru line instead of using the thru provided in the autocal box.

```
:SENSe{1-16}:CORRection:COLLect:ECAL:TRUEthru
```

The command’s input argument list consists of comma separated data with alternating port selections and on/off flags. For instance, if AutoCals are being performed on Ports 1 and 3 and Ports 2 and 4 with true thrus on the Ports 2 and 4 calibration, then the command is:

```
:SENS1:CORR:COLL:ECAL:TRUE PORT13,OFF,PORT24,ON
```

Automatic detection of the Autocal module orientation is available with the following command only with 2-Port configurations, but is not offered when a 4-port test set is connected:

```
:SENSe{1-16}:CORRection:COLLect:ECAL:AUTOmatic:ORIentation[:STATe]
```

Once the AutoCal setup is complete, the following command starts the calibration:

```
:SENSe{1-16}:CORRection:COLLect:ECAL:BEGin?
```

The AutoCal calibration may require interaction with a user to perform some mechanical setup steps such as reversing the autocal box, connecting the autocal box to different port(s), or connecting external thru lines. As each step is completed, send the following command to instruct the VNA to continue with the measurements:

```
:SENSe{1-16}:CORRection:COLLect:ECAL:CONTinue?
```

The following command outputs a copy of the Autocal messages list:

:SENSe{1-16}:CORRection:COLLect:ECAL:MSGS:LIST?

This command outputs a list of up to four comma-separated items from the following list:

0 - AssurancePassed

1 - Update

2 - TrueThru

3 - Adapter

4 - NoModule

5 - NoOrient

6 - NoFile

7 - NoMatch

8 - No12T

9 - NotAllowed

10 - OutOfRange

11 - AssuranceFailed

12 - Aborted

13 - AbortOK

14 - AbortNotOK

15 - ACError

16 - ACFatalError

17 - DoneCalculateCoeff

18 - ACConnectCalB

19 - CharacBad

20 - DisplayMessage

21 - ConnectToPort1

22 - ConnectToPort2

23 - ConnectToPort3

24 - ConnectToPort4

25 - ConnectToPorts12

26 - ConnectToPorts13

27 - ConnectToPorts14

28 - ConnectToPorts23

29 - ConnectToPorts24

30 - ConnectToPorts34

31 - ConnectThrubwPorts12

32 - ConnectThrubwPorts13

33 - ConnectThrubwPorts14

34 - ConnectThrubwPorts23

35 - ConnectThrubwPorts24

36 - ConnectThrubwPorts34

37 - SequentialBegins

The following is a list of AutoCal return codes:

Table 2-23. AutoCal Module Return Code Definitions (1 of 2)

Return Code	Code Description
0	Assurance: Assurance passed for AutoCal Modules that have an assurance step. AutoCal complete for AutoCal Modules that have no assurance step.
1	Update: AutoCal complete for AutoCal Modules that have no assurance step.
2	True Thru: Connect through line.
3	Adapter: Reverse AutoCal module connection for Adapter Removal
4	NoModule: AutoCal module not found.
5	NoOrient: AutoCal module orientation not detected.
6	NoFile: AutoCal Characterization file not found.
7	NoMatch: AutoCal Characterization file and module mismatch. Check AutoCal serial number match to AutoCal Characterization file name.
8	No12T: Characterization function needs Full 2-Port (12-Term) calibration. Full 2-Port calibration not found.
9	NotAllowed: AutoCal automatic orientation not available on Lightning modules. Orientation must be manually specified.
10	OutOfRange: Frequencies are out of AutoCal module range.
11	AssuranceFailed: Assurance failed for AutoCal modules that have an assurance step. Not applicable for AutoCal Modules that do not have an assurance step
12	Aborted: AutoCal calibration or Characterization aborted, typically by user.
13	AbortOK: Abort operation concluded successfully.
14	AbortNotOK: Abort operation not concluded successfully.
15	ACError: AutoCal unspecified error.
16	ACFatalError: AutoCal unspecified fatal error.
17	DoneCalculateCoeff: AutoCal module has completed calculating required coefficients.
18	ACConnectCalB
19	CharacBad: Characterization is bad.
20	DisplayMessage
21	ConnectToPort1: Connect AutoCal module to Port 1.
22	ConnectToPort2: Connect AutoCal module to Port 2.
23	ConnectToPort3: Connect AutoCal module to Port 3. Requires 4-Port VNA.
24	ConnectToPort4: Connect AutoCal module to Port 4. Requires 4-Port VNA.
25	ConnectToPorts12: Connect AutoCal module to Ports 1 and 2.
26	ConnectToPorts13: Connect AutoCal module to Ports 1 and 3. Requires a 4-Port VNA.
27	ConnectToPorts14: Connect AutoCal module to Ports 1 and 4. Requires a 4-Port VNA.
28	ConnectToPorts23: Connect AutoCal module to Ports 2 and 3. Requires a 4-Port VNA.
29	ConnectToPorts24: Connect AutoCal box to Ports 2 and 4. Requires a 4-Port VNA.
30	ConnectToPorts34: Connect AutoCal module to Ports 3 and 4. Requires a 4-Port VNA.
31	ConnectThrubwPorts12: Connect Thru line to Ports 1 and 2.
32	ConnectThrubwPorts13: Connect Thru line to Ports 1 and 3. Requires a 4-Port VNA.
33	ConnectThrubwPorts14: Connect Thru line to Ports 1 and 4. Requires a 4-Port VNA.

Table 2-23. AutoCal Module Return Code Definitions (2 of 2)

Return Code	Code Description
34	ConnectThrubwPorts23: Connect Thru line to Ports 2 and 3. Requires a 4-Port VNA.
35	ConnectThrubwPorts24: Connect Thru line to Ports 2 and 4. Requires a 4-Port VNA.
36	ConnectThrubwPorts34: Connect Thru line to Ports 3 and 4. Requires a 4-Port VNA.

All VectorStar AutoCal modules support an assurance step. Some Lightning AutoCal modules do not support an assurance step.

LRL Calibration

The LRL calibration method must first be specified using the command below:

```
:SENSe{1-16}:CORRection:COLLect LRL
```

FULL3 LRL Calibration

The FULL3 LRL calibration is accomplished by performing two, FULL2 calibrations having a common port. The following command sets the calibration type to a full 3-port LRL calibration for the indicated channel:

```
:SENSe{1-16}:CORRection:COLLect:LRL:PORT{13 | 14 | 23 | 24}:FULL3
```

The port selection for the CALA calibration is limited to the ports shown below. The command's argument must complement that of the port selection for the CALB calibration. The port selections available for the CALB calibration are limited based on the port selection made in the CALA calibration as follows:

CALA Port Selection		CALB Port Choices
PORT13 or PORT24	<-------->	PORT14 or PORT23
PORT14 or PORT23	<-------->	PORT13 or PORT24

Caution	If the rules above for the CALB calibration port pair is violated, the FULL3 command will fail.

FULL4 LRL Calibration

The FULL4 LRL calibration is accomplished by performing two, FULL2 calibrations on independent port pairs. The following command sets the calibration type to a full 4-port LRL calibration for the indicated channel:

```
:SENSe{1-16}:CORRection:COLLect:LRL:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL4
```

In addition, one to four thru lines must be measured using the commands listed below:

```
:SENSe{1-16}:CORRection:COLLect:THRu:CLEar
```

```
:SENSe{1-16}:CORRection:COLLect:THRu:ADD
```

Available throughs are:

```
THRu12 | THRu13 | THRu14 | THRu23 | THRu24 | THRu34
```

Setting Up the Device Parameters

The following commands are then used to set up the device parameters:

The following three commands provide backward compatibility with legacy VectorStar software and are used to set up the frequency, line length, and loss values:

```
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:FREQuency
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LENGth
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LOSS
```

The following three commands provide support for newer VectorStar software and are used to set up the frequency, line length, and loss values:

```
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:FREQuency
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LENGth
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LOSS
```

The following commands provide support for newer VectorStar software and for the 4-port test set. Omitting the optional [:CALa] keyword in these commands provide support for legacy 2-port instruments:

```
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND1:REFLection:TYPe
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND2:REFLection:TYPe
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BANDs:COUNt
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C0
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L0
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFFS
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:R
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:Z0
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:TYPe
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:OPEN:OFFS
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:REFPlane
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:SHORT:OFFS
```

The following commands are used for the corresponding CALB parameters:

```
:SENSe{1-16}:CORRection:COLLect:LRL:CALB{1-5}:REFLection:TYPe
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BANDs:COUNt
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:FREQuency
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LENGth
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LOSS
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C0
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L0
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFFS
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:R
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:Z0
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:TYPe
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:OPEN:OFFS
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:REFPlane
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:SHORT:OFFS
```

The following command defines which match corresponds to which port:

```
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:MATCH:PORT
```

The following commands collect LRL calibration data:

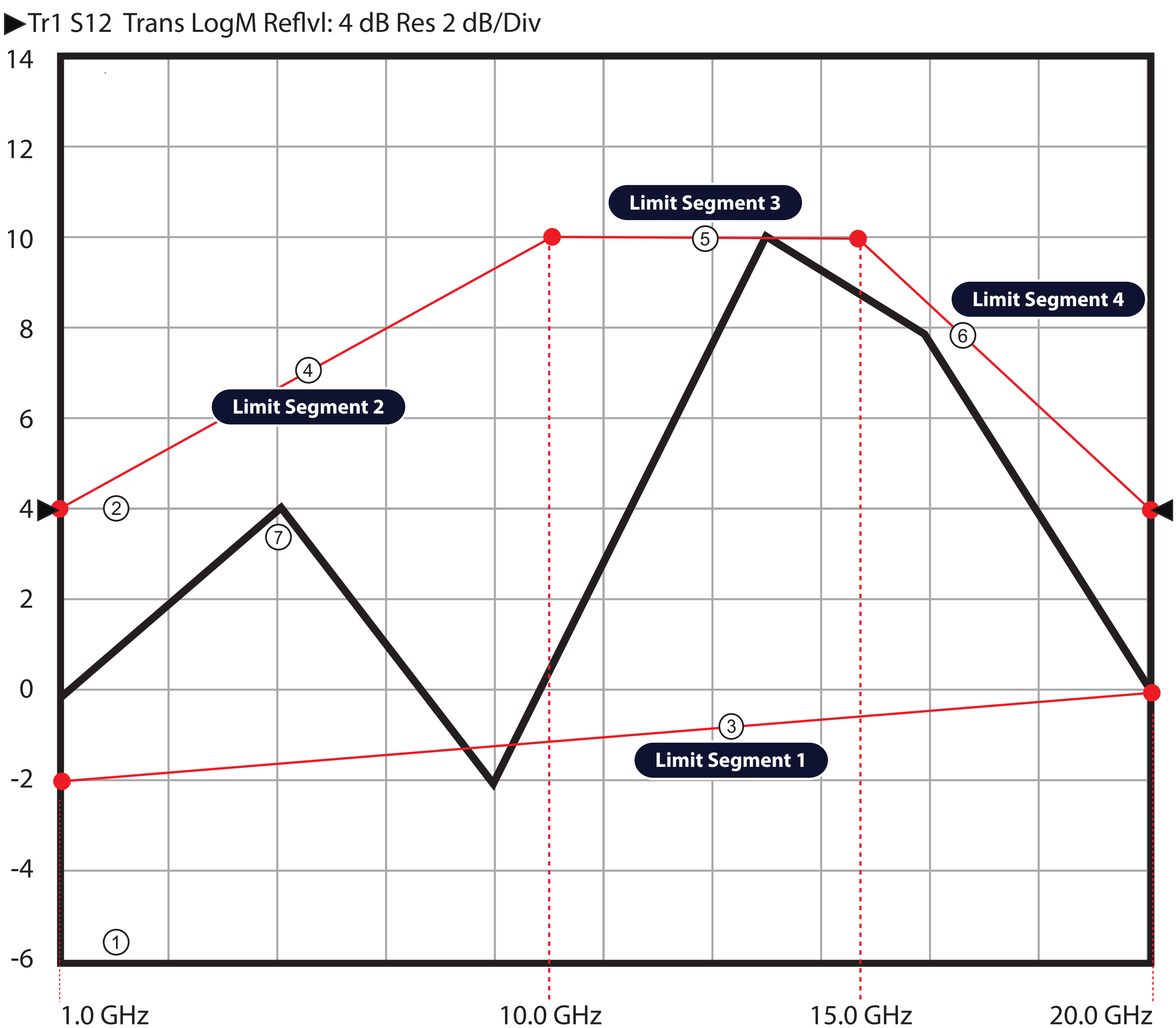
```
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT{1-4}:MATCH
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT{12 | 13 | 14 | 23 | 24 |
34}:LINE
```

2-24 Command Script Example – Limit Lines

This script example provides the basic procedure for establishing upper and lower limit lines for a trace.

Limit Lines for Single Rectilinar Trace Display

The requirements for this limit line example are shown in the figure below where one lower limit has been established with three upper limits.



1. Representative trace display for frequency from 1 GHz to 20 GHz, and levels from –6 to +14 dB.
2. Log Magnitude trace display measuring S12, with:
   - Resolution set at 2 dB per division.
   - Reference value set at 2 dB.
   - Reference position set at 5.
3. Limit Segment 1 – A lower limit line from 1 GHz at –2 dB to 20 GHz at 0 dB.
4. Limit Segment 2 – An upper limit line from 1 GHz at 4 dB to 10 GHz at 10 dB.
5. Limit Segment 3 – An upper limit line from 10 GHz at 10 dB to 15 GHz at 10 dB.
6. Limit Segment 4 – An upper limit line from 15 GHz at 10 dB to 20 GHz at 0 dB.
7. Typical signal trace for this DUT.

Figure 2-6. Limit Line Concept and Example

Required Equipment

- VectorStar MS4644B VNA, K Connectors, 10 MHz to 40 GHz
- Anritsu 36585K-2MF Precision Automatic Calibrator (AutoCal) Module Calibration Kit, K (male) to K (female)

Prerequisites

- The VNA has warmed up for at least 90 minutes.
- The AutoCal Module characterization file has been installed on the VNA.

DUT Requirements

The DUT measurements require the following parameters:

- S-Parameter Required: S12
- Frequency Range: 1 GHz to 20 GHz
- Segment 1: Lower Limit, -2 dB at 1 GHz to 0 dB at 20 GHz
- Segment 2: First Upper Limit, 4 dB at 1 GHz to 10 dB at 10 GHz
- Segment 3: Second Upper Limit, 10 dB at 10 GHz to 10 dB at 15 GHz
- Segment 4: Third Upper Limit, 10 dB at 15 GHz to 4 dB at 20 GHz

Channel and Trace Display Requirements

The following VNA setup parameters are required:

- Channels: 1
- Traces: 1
- Trace Display Type: Log Magnitude
- Trace Scale Resolution: 2 dB/Division
- Trace Scale Reference Value: 4 dB
- Trace Scale Reference Position: 5 (Positions the reference value above at the fifth gridline from the display counting from the display bottom.

VNA General Setup and Configuration

Throughout the script examples, long form commands are used for clarity. The command explanation follows the command. In this section, the VNA is cleared and per-instrument settings established. Optional commands or queries are noted and are presented for clarity.

`:SYSTem:ERRor:CLEar`

Clears the system error queue.

`:SYSTem:POINt:MAXimum 25000`

Set number of measurement points to 25,000 points. The instrument will reboot if the instrument was in 100,000 point mode. If already in 25,000 point mode, no instrument change and no reboot.

`:DISPlay:COLor:RESet`

Resets all colors to normal factory default value. This returns the channel, channel background, trace, limit line, and graticule colors to their default values.

`:DISPlay:COUNt 1`

Sets one (1) channel.

- When in 25,000 point mode, this can be set to 1 (one), 2, 3, 4, 6, 8, 9, 10, 12, or 16 channels.
- If the channel display is set to a non-listed number (5, 7, 11, 13, 14, 15), the instrument is set to the next higher channel number.

`:CALCulate1:PARameter:COUNt 1`

The command sets the number of traces as 1 on Channel 1.

`:DISPlay:SPLit R1C1`

Sets the channel display layout in a Row-by-Column format where channel window display is set to one channel on one row and one column. This is the same as maxmizing a multi-channel display.

`:DISPlay:WINDow1:ACTivate 1`

The command sets the active channel to the indicated channel number.

`:DISPlay:SIZe MAXimum`

Sets the maximum size of the graphic display. Not really needed here for a one channel display, but shown for a multi-channel and/or multi-trace display.

`:CALCulate1:PARameter1:DEFine S12`

The command sets the measurement parameter as S12 for Trace 1.

`:CALCulate1:PARameter1:FORMat  MLOGarithmic`

The command selects the display format as Log Magnitude (MLOGarithmic) for Trace 1 on Channel 1.

Frequency and Sweep Settings

In this section, the required frequency and sweep settings are established.

`:SENSe1:FREQuency:STARt 1.0E9`

Sets start frequency to 1 GHz.

`:SENSe1:FREQuency:STOP 20.0E9`

Sets the stop frequency to 20 GHz.

`:SENSe1:FREQuency:SPAN?`

Optional query. Span is automatically calculated as Stop Frequency minus Start Frequency. The query returns the resulting span in Hertz.

`19.0E9`

Frequency span is 19 GHz.

`:SENSe1:FREQuency:CENTer?`

Optional query. Center frequency is automatically calculated using Stop Frequency and Start Frequency as:
Fc = ((Fstop - Fstart)/2) + Fstart

`10.5E9`

Center frequency is 10.5 GHz

`:SENSe1:SWEep:TYPe LINear`

The command sets the sweep for Channel 1 as Frequency-Based Linear.

`:SENSe1:SWEep:POINt 401`

Sets the number of measurement points for Channel 1 to 401 points.

- The minimum number of points is 2.
- The maximum number of points is limited to the total point instrument mode as 25,000 or 100,000.

Limit Lines Setup

There are several ways to programmatically add limit lines to a trace display. The technique below uses the :CALCulate{1-16}:SELected:LIMit command subsystem in the following general procedure:

1. Create an empty limit line segment by using the :ADD command to create a blank limit.
   - Note that each :ADD command must be followed by the :TYPE and :X1, :X2, :Y1, and :Y2 commands.
   - The first :ADD command creates a limit line that uses the entire frequency range of the instrument.
   - If the :TYPE and :X1, :X2, :Y1, and :Y2 values are not changed, no further limit line segments can be added.
2. Use the :TYPE command to define the segment as a lower or upper limit.
3. Use the :X1 and :X2 commands to configure the horizontal X-Axis start and stop points for each limit segment. In this example, the start and stop values are frequency in GHz.
4. Use the :Y1 and :Y2 commands to configure the vertical Y-Axis start and stop points for each limit segment. In this example, the start and stop values are in dB.
5. The :TYPE and :X1, :X2, :Y1, and :Y2 commands can be issued in any sequence for the segment being defined.

Clear Previous Limit Lines

Best practices recommend clearing all previous segments.

`:CALCulate1:SELected:LIMit:SEGMent:CLEar`

The command clears all the limit segment definitions on the active trace of the indicated channel.

Create and Configure Limit Line Segment 1

In this section, the first limit line is added, and then configured as to limit line type, start and stop frequencies, and start and stop Y-axis parameters.

`:CALCulate1:SELected:LIMit:SEGMent:ADD`

On Channel 1, the command adds a limit line segment. This limit line segment will be identified as Segment 1 and set as the lower limit line across the entire frequency range of interest.

`:CALCulate1:SELected:LIMit:SEGMent1:TYPe LOWer`

Sets the Channel 1 Segment 1 limit line type as a lower limit line.

`:CALCulate1:SELected:LIMit:SEGMent1:X1 1.0E9`

Sets the Channel 1 Segment 1 lower limit line start frequency value at 1 GHz.

`:CALCulate1:SELected:LIMit:SEGMent1:X2 20.0E9`

Sets the Channel 1 Segment 1 lower limit line stop frequency value at 20 GHz.

`:CALCulate1:SELected:LIMit:SEGMent1:Y1 -2.0`

Sets the Channel 1 Segment 1 lower limit start Y1 value at –2.0 dB.

`:CALCulate1:SELected:LIMit:SEGMent1:Y2 0.0`

Sets the Channel 1 Segment 1 lower limit stop Y2 value at 0.0 dB.

Create and Configure Limit Line Segment 2

In this section, the second limit line is added, and then configured as to limit line type, start and stop frequencies, and start and stop Y-axis parameters.

`:CALCulate1:SELected:LIMit:SEGMent:ADD`

On Channel 1, command adds a blank limit line segment. This limit line segment will be later identified as Segment 2 and set as the first upper limit line segment.

`:CALCulate1:SELected:LIMit:SEGMent2:TYPe UPPer`

Sets the Channel 1 Segment 2 limit line type as an upper limit line.

`:CALCulate1:SELected:LIMit:SEGMent2:X1 1.0E9`

Sets the Channel 1 Segment 2 upper limit start frequency value at 1 GHz.

`:CALCulate1:SELected:LIMit:SEGMent2:X2 10.0E9`

Sets the Channel 1 Segment 2 upper limit line stop frequency value at 10 GHz.

`:CALCulate1:SELected:LIMit:SEGMent2:Y1 4.0`

Sets the Channel 1 Segment 2 upper limit start Y1 value at 4.0 dB.

`:CALCulate1:SELected:LIMit:SEGMent2:Y2 10.0`

Sets the Channel 1 Segment 2 upper limit stop Y2 value at 10.0 dB.

Create and Configure Limit Line Segment 3

In this section, the third limit line is added, and then configured as to limit line type, start and stop frequencies, and start and stop Y-axis parameters.

```
:CALCulate1:SELected:LIMit:SEGMent:ADD
```

On Channel 1, the command adds a blank limit line segment. This limit line segment will be later identified as Segment 3 and set as the second upper limit line segment.

```
:CALCulate1:SELected:LIMit:SEGMent3:TYPe UPPer
```

Sets the Channel 1 Segment 3 limit line type as an upper limit line.

```
:CALCulate1:SELected:LIMit:SEGMent3:X1 10.0E9
```

Sets the Channel 1 Segment 3 upper limit start frequency value at 10 GHz.

```
:CALCulate1:SELected:LIMit:SEGMent3:X2 15.0E9
```

Sets the Channel 1 Segment 3 upper limit line stop frequency value at 15 GHz.

```
:CALCulate1:SELected:LIMit:SEGMent3:Y1 10.0
```

Sets the Channel 1 Segment 3 upper limit start Y1 value at 10.0 dB.

```
:CALCulate1:SELected:LIMit:SEGMent3:Y2 10.0
```

Sets the Channel 1 Segment 3 upper limit stop Y2 value at 10.0 dB.

Create and Configure Limit Line Segment 4

In this section, the third limit line is added, and then configured as to limit line type, start and stop frequencies, and start and stop Y-axis parameters.

```
:CALCulate1:SELected:LIMit:SEGMent:ADD
```

On Channel 1, command adds a blank limit line segment. This limit line segment will be later identified as Segment 4 and set as the third and final upper limit line segment.

```
:CALCulate1:SELected:LIMit:SEGMent4:TYPe UPPer
```

Sets the Channel 1 Segment 4 limit line type as an upper limit line.

```
:CALCulate1:SELected:LIMit:SEGMent4:X1 15.0E9
```

Sets the Channel 1 Segment 4 upper limit start frequency value as 15 GHz.

```
:CALCulate1:SELected:LIMit:SEGMent4:X2 20.0E9
```

Sets the Channel 1 Segment 4 upper limit line stop frequency value at 20 GHz.

```
:CALCulate1:SELected:LIMit:SEGMent4:Y1 10.0
```

Sets the Channel 1 Segment 4 upper limit start Y1 value at 10.0 dB.

```
:CALCulate1:SELected:LIMit:SEGMent4:Y2 4.0
```

Sets the Channel 1 Segment 4 upper limit stop Y2 value at 4.0 dB.

Configure AutoCal Calibration

For this example, the Anritsu 36585K Precision Automatic Calibrator (AutoCal) Calibration Module will be used to perform the calibration. If the characterization file for the AutoCal module has not been loaded, best practices recommand using the User Interface menus to load the characterization file.

`:SENSe1:CORRection:COLLect:ECAL:AUTOmatic:ORIentation:STATe OFF`

Turn the AutoCal module automatic orientation detection off for Channel 1.

`:SENSe1:CORRection:COLLect:ECAL:ORIentation L1R2`

Set the AutoCal module orientation detection off and sets the port-to-port orientation manually for Channel 1 so that Port 1 is on the leftt and Port 2 is on the right.

`:SENSe1:CORRection:COLLect:ECAL:TRUEthru OFF`

The command turns off the use of the AutoCal True Thru feature, where a cable through is used during the AutoCal calibration for Channel 1. By setting this to OFF, the AutoCal module will use its Internal Thru capability to complete the calibration.

Ready for Measurements

The VNA is ready for measurements.

Chapter 3 — IEEE Commands

3-1 Introduction

This chapter contains all of the IEEE commands that are implemented in the instrument.

Note	When operating the VectorStar VNA through remote programming, the front panel user interface and controls are disabled. To return to local front panel control, press the front panel Clear/Tab key [**Clr -->|**], keyboard **Esc** key, or send the `RTL` command. For general information about GPIB, refer to Section 1-5 “IEEE 488 GPIB Description”.

3-2 Command Descriptions

IEEE commands are used to control instrument status registers, status reporting, synchronization, data storage, and other common functions. All IEEE 488.2 commands are identified by the leading asterisk in the command word and are fully defined in IEEE 488.2.

Each IEEE command is followed a complete descriptive listing of the command description, parameters, and output. If applicable, an example for each command, the default value, and the range information is written out at the end of the individual description.

- See Chapter 2, “Programming the VectorStar Series VNA”, “Notational Conventions” on page 2-7 for definitions of parameters and notations.
- Detailed descriptions of parameter types is available in “Data Transmission Methods” on page 2-11and through the links below.
  - <NR1>
  - <NR2>
  - <NR3>
  - <NRf>
  - <string>
  - <ASCII> or <Arbitrary ASCII>
  - <block> or <arbitrary block>
  - <char>
  - <char1>,<char2>
  - <char1>,<char2>,<char3>
  - <char1>,<char2>,<char3>,<char4>
  - MPND
  - MPNF
  - MPNI

3-3 Numeric Limits

The following numeric limits are abbreviated in the IEEE command descriptions:

- **MPND –** Maximum Positive/Negative Double Precision Number
  - +/– 1.792 693 134 86 E+308
- **MPNI –** Maximum Positive/Negative Integer
  - – 2 147 483 648 to +2 147 483 647
  - +/– 2 E31
- **MPNF –** Maximum Positive/Negative Float Number
  - +/– 3.402 819 E+38

3-4 IEEE 488.2 Commands

*CLS

Description: Clear Status Command. Clears the Status Byte, the Data Questionable Event Register, the Standard Event Status Register, the Standard Operation Status Register, the error queue, the OPC pending flag, and any other registers that are summarized in the Status Byte. No Query

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Syntax Example: `*CLS`

*DDT <Arbitrary Block> | <String>
*DDT?

Description: Define Device Trigger. The command enters the 488.2 Define Device Trigger command with an input of arbitrary block or string. The query returns the 488.2 Define Device Trigger command as an arbitrary block output string. Note that the IEEE488.2 Standard specifies the input type to only be <Arbitrary Block>. In addition, VectorStar VNAs will also accept a <String>.

Cmd Parameters: <Arbitrary Block> | <String>

Query Parameters: NA

Query Output: <Arbitrary Block>

Range: NA

Default: NA

Syntax Example: `*DDT <String>`

`*DDT?`

*ESE <NRf>
*ESE?

Description: Standard Event Status Enable and Query. The command sets the Standard Event Status Enable Register bits. The binary weighted <NRf> data parameter used with this command must have a value between 0 to 255. The query returns the value of the Standard Event Status Enable Register in <NR1> format. Refer to "Status System Reporting" on page 2-29.

Cmd Parameters: <NRf>

Query Parameters: NA

Range: 0 to 255

Query Output: <NR1>

Syntax Example: `*ESE <NRf>`

`*ESE?`

*ESR?

Description: Standard Event Status Register Query. Query only. Returns the value of the Standard Event Status Register in <NR1> format. This command clears the Standard Event Status Register. Refer to "Status System Reporting" on page 2-29.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1>

Range: NA

Default: NA

Syntax Example: `*ESR?`

*IDN?

Description: Identification Query. Query only. This query returns an instrument identification string in IEEE-488.2 specified <Arbitrary ASCII> format consisting of four fields separated by commas. The fields are: <Manufacturer>, <Model>, <Serial #>, <Firmware Revision Level> where the actual model number, serial number, and firmware version of the MS464xB Series VNA queried will be passed. The character output is of indeterminate length and must be the last statement issued if multiple commands and/or queries are issued at the same time. See Chapter 2 for definition of <ASCII>.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <Arbitrary ASCII>

Range: NA

Default: NA

Syntax Example: `*IDN?`

*IST?

Description: Individual Status Query. Output the value of the IST message. This command is not supported by VectorStar VNA hardware.

*OPC

Description: Operation Complete Command. When the *OPC command is encountered, it does nothing. Program flow is allowed to proceed to the next command in the input buffer. No query. Note that *OPC and *OPC? are not a command/query pair although they appear to be.

When the *OPC command is encountered, it immediately sets the Operational Complete bit in the Standard Events Status Register. The *OPC command is a control command for overlapped commands in systems that support command overlapping. The VectorStar does not support overlapped commands, and as a result, no pause function is provided.

Overlapped Command Background

Some non-VectorStar GPIB implementations allow commands to execute simultaneously which is called an Overlapped Command. For example, a calibration step which takes many minutes could be set into operation and then allow other GPIB communication and control commands to proceed while the calibration step continues to take place.

At some point in these non-VectorStar GPIB implementations, the separate Overlapped Operational Streams need to be brought back together and placed under programmatic control in a process called Regaining Synchronization between the controller PC and the instrument. The commands *OPC, *OPC?, and *WAI are designed to regain synchronization control and three different synchronization methods.

IEEE 488.2 Overlapped Command Definitions

The *OPC, *OPC?, and *WAI commands provide coverage for command completion before the next command is parsed and executed and comprise a class of commands termed Overlapped Commands. For instruments that support overlapped commands, each command works differently. Per the IEEE 488.2 specification:

- “IEEE 488.2 defines a distinction between overlapped and sequential commands.
- As defined in IEEE 488.2, a sequential command is one which finishes executing before the next command starts executing. An overlapped command is one which does not finish executing before the next command starts executing.
- These types of commands are described in IEEE 488.2, section 12. Examples are given in IEEE 488.2, Appendix B.
- IEEE 488.2 defines three common commands (*OPC, *OPC?, *WAI) which a device controller can use to synchronize its operation to the execution of overlapped commands.
- Each overlapped command has associated with it a Pending Operation flag.
- The device sets this flag TRUE when it passes the corresponding command from the Execution Control block to the Device Action block.
- The device sets the flag false when the device operation is finished, or has been aborted.”

*OPC Synchronization

The *OPC Operation Complete command is defined in **IEEE 488.2-1992, Section 10.18**. *OPC works only if one or more preceeding commands are overlapped command. In systems that support overlapped commands, when this command is encountered, it waits until all overlapped commands have completed. Once all prior overlapped commands are complete, it instructs the parser to execute the next following command. A typical use would be to issue a *OPC before issuing a command for a long duration measurement sweep. *OPC causes the instrument to continuously sense the No Operation Pending flag. When the No Operation Pending (NOP) flag becomes TRUE, the OPC event bit in the Standard Event Status Register is set to “1” to indicate that the state of all pending operations have been completed. If this bit had been previously programmed to send a Service Request, then the controller would be aware that all is synchronized.

- For example, OV1 through OV3 are overlapped commands where the command series is OV1, OV2, OV3, *OPC, XXX.
- Commands OV1, OV2, OV3 start running. The NOP is set to FALSE.
- Parser execution stops at *OPC and the parser waits while the NOP is FALSE.
- Commands OV1, OV2, and OV3 continue to run in overlapped mode.
- Commands OV1, OV2, and OV3 are finally complete.
- The NOP flag is set to TRUE and *OPC sets the status bit to 1.
- Parser execution resumes.
- Command XXX starts running.

*OPC? Synchronization

In systems that support overlapped commands, the *OPC? Operation Complete Query command waits until all overlapped commands have been completed. When that synchronizing moment arrives, the ASCII character “1” is placed in the output buffer. This sets the MAV bit in the Status byte to TRUE which indicates there is data in the output buffer. If the controller had been attempting to read data, the moment the “1” appears in the output buffer, it is read by the controller making it aware that all is synchronized. When that synchronizing moment arrives, the next command in the input buffer is executed and the command execution is synchronized.

- For example, OV1 through OV3 are overlapped commands where the command series is OV1, OV2, OV3, *OPC?, XXX.
- Commands OV1, OV2, OV3 start running. The NOP is set to FALSE.
- Parser execution stops at *OPC? and the parser waits while the NOP is FALSE.
- Commands OV1, OV2, and OV3 continue to run in overlapped mode.
- Commands OV1, OV2, and OV3 are finally complete.
- The NOP flag is set to TRUE and *OPC? puts a one (“1”) in the output buffer.
- Parser execution resumes.
- Command XXX starts running.

*WAI Synchronization

In systems that support overlapped commands, the *WAI Wait-to-Continue Command causes the parser to wait until all overlapped commands have been completed. When that synchronizing moment arrives, the next command in the input buffer is executed and command execution is synchronized.

- For example, OV1 through OV3 are overlapped commands where the command series is OV1, OV2, OV3, *WAI, XXX.
- Commands OV1, OV2, OV3 start running. The NOP is set to FALSE.
- Parser execution stops at *WAI and the parser waits while the NOP is FALSE.
- Commands OV1, OV2, and OV3 continue to run in overlapped mode.
- Commands OV1, OV2, and OV3 are finally complete.
- The NOP flag is set to TRUE.
- Parser execution resumes.
- Command XXX starts running.

VectorStar and Overlapped Commands

In VectorStar VNAs, and in Anritsu’s earlier 360 and Lightning VNAs, there are no Overlapped Commands. Everything works synchronously and no command is executed until the command which preceded it has finished. Because of this, the commands *OPC, *OPC? and *WAI work slightly differently than the IEEE 488.2 specification.

Related Commands: *OPC, *OPC?, and *WAI commands.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1>

Range: NA

Default: NA

Syntax Example: `*OPC`

*OPC?

Description: Operation Complete Query. Query only. Not a command/query pair with *OPC. When the *OPC? command is encountered, it does nothing. Program flow is allowed to proceed to the next command in the input buffer.

Per IEEE 488.2, *OPC? is an Overlapped Command which VectorStar VNAs do not support. When the *OPC? command is encountered, it immediately puts the character “1” (one) in the Output Queue buffer, and parser execution continues. It sets the MAV bit true when all pending operations are complete.

The *OPC, *OPC?, and *WAI Overlapped Commands are related. Because VectorStar does not support overlapped commands, they do not function exactly as specified in IEEE 488.2. See the description of Overlapped Commands in the *OPC command description above.

Related commands: *OPC, *OPC?, and *WAI commands.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1>

Range: NA

Default: NA

Syntax Example: `*OPC?`

*OPT?

Description: Operation Query. Query only. The query reads out the identification number of an option installed in the MS4640B. See Chapter 2 for definition of <ASCII>.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default: NA

Query Output: <Arbitrary ASCII>

Syntax Example: `*OPT?`

*PRE <NRf>
*PRE?

Description: The command enters the 488.2 Parallel Poll Register Enable mask. The query outputs the 488.2 Parallel Poll Register Enable mask. This command/query is not supported by VectorStar VNA hardware.

*RST

Description: Reset Command. The *RST command performs a device reset of the MS4640B VNA to a pre-defined condition or to a user-defined condition. No query. The user-defined condition of *RST resets all user programmable parameters to those defined by the user in a saved configuration file. The pre-defined condition of the *RST command sets the defaults described below. For additional information on default parameter values, see each SCPI command in this manual.

*RST Does Reset the Following Parameters

Except as explicitly excluded in the next section, the *RST command does the following:

- Sets the device-specific functions to a known state that is independent of the past-use history of the device;
- Device specific commands may be provided to program a different reset state than the original factory-supplied one;
- Sets the macro defined by *DDT to a device-defined state;
- Disables macros;
- Forces the device into the OCIS state (Operation Complete Command Idle State);
- Forces the device into the OQIS state (Operation Complete Query Idle State).

*RST Does Not Reset the Following Parameters

The *RST command does not change the parameters listed below:

- Does not change the state of the IEEE 488.1 interface;
- Does not change the selected IEEE 488.1 address of the device;
- Does not change the GPIB address of the device;
- Does not change the Output Queue;
- Does not change any Event Enable Register settings including the Standard Event Status Enable Register;
- Does not change any Event Register setting including the Standard Event Status Register settings;
- Does not change the power-on-status-clear flag setting;
- Does not change the Service Request Enable Register;
- Does not change the maximum number of instrument points to 25,000 if it is set to 100,000 nor change them to 100,000 if it is set to 25,000.

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `*RST`

*SRE <NRf>
*SRE?

Description: Service Request Enable. The command sets the Service Request Enable Register bits. A zero value in the command resets the register. The query returns the value of the Service Request Enable Register in <NR1> format. Bit 6 is always zero. The integer data parameter used with this query have a value between 0 to 255.

Cmd Parameters: <NRf>

Query Parameters: NA

Query Output: <NR1>

Range: 0 to 255; 0 performs a register reset.

Default: NA

Syntax Example: `*SRE <NRf>`

`*SRE?`

*STB?

Description: Read Status Byte Query. Query only. Returns the content of the Status Byte Register (bits 0 through 5 and 7). Bit 6 is the Master Summary Status bit value. This command does not reset the status byte values.

Query Parameters: NA

Query Output: <NR1>

Range: NA

Default: NA

Syntax Example: `*STB?`

*TRG

Description: Trigger Command. Triggers the instrument if :TRIGger:SOURce command data parameter is set to REMOTE. Performs the same function as the Group Execute Trigger (<GET>) bus command as defined in IEEE 488.2. *TRG or the <GET> can trigger a measurement. No query.

The *TRG and <GET> commands are handled differently by the parser. The *TRG command:

- The *TRG command can be mixed with other GPIB commands in a command string which goes through the parser.
- The *TRG command can only trigger a single measurement event on only one device in the GPIB network.

The <GET> command:

- The <GET> gets turned into the command *TRG and goes through the parser by itself.
- A <GET> can trigger single measurement events on any or all instruments in the GPIB network.

What is measured depends on the setting of the *DDT and :TRIGger[:SEQuence]:EXTernal:TYPe commands.

- If a *DDT command has been issued previously, the *TRG or <GET> will execute what is defined in the *DDT instead.
- If a *DDT has not been issued, the *TRG triggers a measurement based on the :TRIG:EXT:TYP command which can be measurement of a point, a single sweep, a single channel, or all channels.
- If a sweep is selected, on 2-Port VNAs, it can be either a either a forward or reverse sweep. On 4-Port VNAs, the sweep can be between any two ports.

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Output: NA

Syntax Example: `*TRG`

*TST?

Description: Self Test Query. Query only. Performs a self test and outputs the self test status in <NR1> format with the following values:

- 0 indicates that self test passed
- Any number greater than 0 indicates the number of the self test that failed
- 144 indicates that the self test was aborted.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1>

Range: NA

Default: NA

Syntax Example: `*TST?`

*WAI

Description: Wait-to-Continue Command. When the *WAI command is encountered, it does nothing. Program flow is allowed to proceed to the next command in the input buffer. No query.

The *WAI command is an Overlapped Command that provides coverage for command completion when the device supports overlapped command execution. The VectorStar does not support overlapped commands, and as a result, the *WAI does not function exactly as specified in IEEE 488.2. See the description of Overlapped Commands in the *OPC command description above.

Related commands: *OPC, *OPC?, and *WAI.

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Syntax Example: `*WAI`

Chapter 4 — System and Troubleshooting Commands

4-1 Introduction

This chapter provides a listing and description of general system-related commands used for configuration, diagnostics, internal calibration, and troubleshooting. Complete details of each command is given in the listing following each command SCPI subsystem. If applicable, an example for each command, the default value, and the SCPI information is written out at the end of the individual description.

Note	When operating the VectorStar VNA through remote programming, the front panel user interface and controls are disabled. To return to local front panel control, press the front panel Clear/Tab key [**Clr -->|**], keyboard **Esc** key, or send the `RTL` command. For general information about GPIB, refer to "IEEE 488 GPIB Description" on page 1-3.

4-2 Parameters and Notations

See Chapter 2, "Notational Conventions" on page 2-7 for definitions of parameters and notations. A notation summary table is available in Table 2-4, "Parameter Notations" on page 2-8.

Detailed descriptions are available in "Data Transmission Methods" on page 2-11 and through the links below.

- <NR1>
- <NR2>
- <NR3>
- <NRf>
- <string>
- <ASCII> or <Arbitrary ASCII>
- <block> or <arbitrary block>
- <char>
- <char1>,<char2>
- <char1>,<char2>,<char3>
- <char1>,<char2>,<char3>,<char4>
- MPND
- MPNF
- MPNI

4-3 Numeric Limits

The following numeric limits are abbreviated in the command descriptions:

- **MPND**
  - Maximum Positive/Negative Double Precision Number
  - +/– 1.792 693 134 86 E+308
- **MPNI**
  - Maximum Positive/Negative Integer
  - – 2 147 483 648 to +2 147 483 647
  - +/– 2 E31
- **MPNF**
  - Maximum Positive/Negative Float Number
  - +/– 3.402 819 E+38

4-4 System Commands

This section provides a listing of general system-related commands used for configuration, diagnostics, internal calibration, and troubleshooting.

ALCCALON <char>
ALCCALON?

Description: ALC Calibration On/Off. The command turns ALC Calibration on/off. The query outputs on/off status of ALC Calibration.

Output: NA

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Command Type: System Command

FIND <string>

Description: Find Matching Help Lines. The command outputs help lines containing a matched string. No query.

Output: <Arbitrary Block>

Parameters: NA

Command Type: System Command

FRCVCALON <char>
FRCVCALON?

Description: Factory Receiver Calibration On/Off. The command turns the factory receiver calibration on/off. The query outputs the factory receiver calibration on/off status.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Command Type: System Command

FRFCALON <char>
FRFCALON?

Description: Factory RF Calibration On/Off. The command turns the factory RF calibration on/off. The query outputs the factory receiver RF calibration on/off status.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0 |

Command Type: System Command

HELP <char>

Description: Help Output. Output help fields on a particular mnemonic. No query.

Cmd Parameters: <char>Character string as required of mnemonic of interest.

Output: <Arbitrary Block>

Parameters: NA

Command Type: System Command

IRCVCAL <arbitrary block>
IRCVCAL? <char>

Description: Input Receiver Calibration Data and Status. The command inputs receiver calibration data. The query outputs the parameter channels receiver cal existence status.

Cmd Parameters: <arbitrary block>

Query Parameters: <char> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

Query Output: <char> 1 | 0

Command Type: System Command

ISDEBUG?

Description: Query only. The query returns the VectorStar VNA Debug Mode status.

Cmd Parameters: NA

Query Output: <NR1>

Command Type: Troubleshooting Command

ISRCVCAL?

Description: Query only. Output is the parameter channels receiver cal existence status.

Query Parameters: <char> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

Query Output: <char> 1 | 0

Command Type: System Command

LANG <char>
LANG?

Description: Parser Language Set. Sets the parser language. The following parser languages are available:

- Native
- Lightning
- HP8510

Native Language Set

This sets the language to Native SCPI. The following additional items are changed:

- The status byte structure changes to the *Native* status byte structure.
- The IEEE 488.2 arbitrary block header format is changed such that the length of the header is always 11 characters long regardless of the size of the arbitrary block.
- The ASCII output delimiting mode is change to *Enhanced*, where individual items of an output response are separated by line feeds or alternating commas and line feeds.

Lightning Language Set

This sets the language to Lightning emulation. The following addition items are changed:

- The Status byte structure changes to the Lightning status byte structure
- The IEEE 488.2 Arbitrary block header format is changed such that the length of the header changes based on the size of the arbitrary block.
- The ASCII output delimiting mode is changed to *Normal*. This means that ASCII output data, which would normally be separated by commas, remains in ASCII data format.
- Markers are forced into the *Coupled* mode.
- The response to the Lightning query OID is changed to Lightning equivalent model numbers and frequency ranges. For example, the following queries and responses:
- MS4642B returns '37347D,0.040000,20.000000,-20.00,0.00,005.06'
- MS4644B returns '37369D,0.040000,40.000000,-20.00,0.00,005.06'
- MS4645B returns '37377D,0.040000,50.000000,-20.00,0.00,005.06'
- MS4647B returns '37397D,0.040000,65.000000,-12.00,0.00,005.06'

HP8510 Language Set

This sets the language to HP8510 Emulation. The following additional items are changed:

- The status byte structure changes to the HP8510 status byte structure.
- The IEEE 488.2 arbitrary block is replaced by an ASCII/binary format unique to the HP8510.
- Markers are forced into the *Coupled* mode.

Cmd Parameters: <char> LIGHTning | HP8510 | NATive

Query Parameters: <char> LIGHT | HP8510 | NAT

Command Type: System Command

LIST

Description: List All Mnemonics. The command outputs a list of all mnemonics. No query.

Output: <Arbitrary Block>

Cmd Parameters: NA

Command Type: System Command

ONCS

Description: Output Next Calibration Step String. The command outputs a string representing the next calibration step. No Query

Output: <string>

Parameters: NA

Command Type: System Command

ORCVCAL

Description: Output Receiver Calibration Data. The command outputs the receiver calibration data. No Query

Output: <Arbitrary Block>

Parameters: NA

Command Type: System Command

RCVCAL <char1>, <char2>

Description: Receiver Calibration. The command performs a Receiver Calibration for the indicated parameter on the indicated port. No Query

Cmd Parameters: <char1>: A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

<char2>: PORT1 | PORT2 | PORT3 | PORT4

Command Type: System Command

RCVCALON <char1>, <char2>
RCVCALON? <char1>

Description: Receiver Calibration Parameter Channels On/Off and Status. The command toggles the parameter channels receiver cal on or off. The query outputs the parameter channels receiver cal on/off status.

Cmd Parameters: <char1>: A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

<char2>: 1 | 0

Query Parameters: <char1>: A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

<char2>: 1 | 0

Command Type: System Command

RTL

Description: Send all devices to local operation. No query.

Output: NA

Parameters: NA

Command Type: System Command

TST?

Description: Self Test and Output Status. Query only. The query performs an instrument self test and outputs its status.

Query Output: <NR1>

Parameters: NA

Command Type: System Command

TSTAIF?

Description: Analog IF Self Test and Output Status. Query only. The query performs an analog IF self test and outputs its status.

Output: <NR1>

Parameters: NA

Command Type: System Command

TSTDECK?

Description: RF Deck Self Test and Output Status. Query only. The query performs an RF deck self test and outputs its status.

Parameters: NA

Command Type: System Command

TSTDSP?

Description: DSP/PC Self Test and Output Status. Query only. The query performs a DSP/PC self test and outputs its status.

Query Output: <NR1>

Parameters: NA

Command Type: System Command

TSTRES?

Description: Self Test Results Output. Query only. The query outputs the last self test results.

Query Output: <Arbitrary Block>

Parameters: NA

Command Type: System Command

TSTSRC?

Description: Source Self Test and Output Status. Query only. The query performs a source self test and outputs its status.

Query Output: <NR1>

Parameters: NA

Command Type: System Command

:SYSTem:TIMe:ELAPsed:INITial?

:SYST:TIM:ELAP:INIT?

Outputs the number of seconds since the initial VNA turn on

:SYSTem:TIMe:ELAPsed:TURNon?

:SYST:TIM:ELAP:TURN?

Outputs the number of seconds since the most recent VNA turn on

:SYSTem:TIMe:INITial?

:SYST:TIM:INIT?

Outputs the initial VNA turn on time

:SYSTem:TIMe:TURNon?

:SYST:TIM:TURN?

Outputs the most recent VNA turn on time

Chapter 5 — SCPI Commands

5-1 Introduction

This chapter co5ntains all of the SCPI commands (required and native) that are implemented in the instrument. The SCPI commands are grouped by their respective subsystems. For each subsystem, the commands described in detail in the listing. The notation corresponds to one of the SCPI standards to a large extent.

Note	When operating the VectorStar VNA through remote programming, the front panel user interface and controls are disabled. To return to local front panel control, press the front panel Clear/Tab key [**Clr -->|**], keyboard **Esc** key, or send the `RTL` command. For general information about GPIB, refer to Section 1-5 "IEEE 488 GPIB Description".

5-2 Minimum/Maximum Frequency Limits and Related Parameters

The minimum and maximum instrument frequencies depend on the instrument model, the installed options, and whether the VNA is a standalone unit or part of a system. See Chapter 1, "General Information", "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for additional information. The following tables for standalone VNA frequency limits are available:

- Table 1-4, "Standalone VNAs – Default Start, Default CW, and Default Stop Frequencies" on page 1-23
- Table 1-5, "Standalone VNAs – Minimum Start, Minimum CW, and Maximum Start Frequencies" on page 1-24
- Table 1-6, "Standalone VNAs – Minimum Stop, Maximum Stop, and Maximum CW Frequencies" on page 1-25
- Table 1-7, "Standalone VNAs – Default Frequency Span and Maximum Frequency Span" on page 1-26
- Table 1-8, "Standalone VNAs – Minimum Center Frequency and Maximum Center Frequency" on page 1-27
- Table 1-9, "Standalone VNAs – Default Center Frequencies" on page 1-28
- For general information on frequency limits for VectorStar VNAs in systems, also in Chapter 1, "General Information", see:
- "VNA Systems with ME7828A Configured as Broadband System" on page 1-28
- "VNA Systems with ME7828A Configured as a Millimeter-Wave System" on page 1-28
- "VNA Systems with ME7828A Using Multiple Source" on page 1-28
- "VNA Systems with ME7838x Modular Broadband System" on page 1-28

5-3 Command Level Hierarchy

The different levels of the SCPI command hierarchy are represented in a table by means of indentations to the right. Lower command levels are indented farther to the right. Observe that the complete notation of the command always includes the higher levels as well.

For example, `:SENSe{1-16}:FREQuency:CENTer` has three command levels and is represented in the table as follows:

```
:SENSe{1-16} (first level)
   :FREQuency (second level)
      :CENTer (third level)
```

The maximum number of command levels is eight. For example, the command:
`:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-4}:PORT12:LINE:FREQuency`
has eight command levels as follows:

```
:SENSe{1-16} (first level)
   :CORRection (second level)
      :COLLect (third level)
         :LRL (fourth level)
            :DEVice{1-4} (fifth level)
               :PORT12 (sixth level)
                  :LINE (seventh level)
                     :FREQuency (eighth level)
```

Command Descriptions and Notation Conventions

Complete details of each command is given following the table of commands for each SCPI subsystem. If applicable, an example for each command, the default value, and the SCPI information is written out at the end of the individual description.

Numeric Limits

The following numeric limits are abbreviated in the SCPI command descriptions:

- MPND – Maximum Positive/Negative Double Precision Number

  +/– 1.792 631 348 6 E+308

- MPNI – Maximum Positive/Negative Integer

  – 2 147 483 648 to +2 147 483 647

  +/– 2 E31

- MPNF – Maximum Positive/Negative Float Number

  +/– 3.402 819 E+38

Notational Conventions

See Chapter 2, “Programming the VectorStar Series VNA”, “Notational Conventions” on page 2-7 for definitions of parameters and notations.

Detailed descriptions of parameter types is available in “Data Transmission Methods” on page 2-11 and through the links below.

- <NR1>
- <NR2>
- <NR3>
- <NRf>
- <string>
- <ASCII> or <Arbitrary ASCII>
- <block> or <arbitrary block>
- <char>
- <char1>,<char2>
- <char1>,<char2>,<char3>
- <char1>,<char2>,<char3>,<char4>
- MPND
- MPNF
- MPNI

5-4 General Parameters

The following general parameters are defined for multiple subsystems in the sections following.

- :CALCulate{1-16} refers to the indicated channel.
  If the index number is not used, the command applies to index 1.
- :CALibration{1-2} refers to the indicated calibration file to be merged.
  If the index number is not used, the first calibration file is used.
- :CONTrol{1-16} refers to the indicated channel number.
  If the index number is not used, the command applies to index 1.
- :DEVice{1-4} refers to the indicated device for the LRL calibration on the indicated channel.
  If the index number is not used, the first device is used.
- :EXTernal{1-4} refers to the indicated external source.
  If the index number is not used, the command applies to the external source 1.
- :FILe{1-4} refers to one of the four hybrid calibration files. Each file must be uniquely identified.
  If the index number is not used, the command displays a syntax error.
- :FLAT{1-2} refers to the flat test port power flatness coefficients for the indicated port and the active channel. If the index number is not used, the command is applied to Port 1 on the active channel.
- :FSEGMent refers to the active frequency-based segment.
- :FSEGMent{1-50} refers to the indicated frequency-based segment.
  If the index number is not used, the command applies to the frequency-based segment 1.
- :GENerator{1-4} refers to indicated internal pulse generator.
  If the index number is not used, the command applies to the internal pulse generator 1.
- :INTernal{1-2} refers to indicated internal source.
  If the index number is not used, the command applies to the internal source 1.
- :ISEGMent refers to the active index-based segment.
- :ISEGMent{1-50} refers to the indicated index-based segment.
  If the index number is not used, the command applies index-based segment 1.
- :LINearity{1-2} refers to the power sweep linearity coefficients for the indicated port and the active channel. If the index number is not used, the command is applied to Port 1.
- :MARKer refers to the active marker.
- :MARKer{1-13} refers to the indicated marker where Marker 1 through 12 are standard measurement markers and Marker 13 is the reference marker. If the index number is not used, the command applies to the marker 1.
- :OFFSet{1-50} refers to the indicated multiple source band.
  If the index number is not used, the command applied to offset band 1.
- :OPERand{1-2} refers to either the first or second operand on the active trace.
  If the index number is not used, operand refers to operand 1.
- :PARameter{1-16} refers to the indicated trace.
  If the index number is not used, the command applies to trace 1.
- :PORT{12 | 13 | 14 | 23 | 24 | 34} refers to the indicated port pair. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.
  If the index number is not used, the command is applied to Port 1 and Port 2.
- :PORT{13 | 14 | 23 | 24} refers to the indicated port pair such as Port23 refers to Port 2 and Port 3. Best practices recommend stating the port pair for every command. The use of these commands require a 4-Port VNA instrument.
  If the index number is not used, the command applies to Port 1 and Port 3.
- :PORT{123 | 124 | 134 | 234} refers to the indicated port triplet to be used in the hybrid calibration. A 4-Port VNA instrument is required.
  If the index number is not used, the command is applied to Port 1, Port 2, and Port 3.
- :PORT{1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34 | 123 | 124 | 134 | 234 | 1234} refers to the indicated individual port, port pair, port triplet, or port quartet to be used in the calibration. The use of Port 3 and/or Port 4

requires a 4-Port VNA instrument and related test set.
If the index number is not used, the command applies to Port 1.

- :PORT{1-4} refers to the indicated port. The use of Port 3 or Port 4 requires a 4-Port VNA instrument.
If the index number is not used, the command is applied to Port 1.
- :SEGMent refers to the currently active limit line.
- :SEGMent{1-50} refers to the indicated limit line.
If the index number is not used, the command applies to segment 1.
- :SENSe{1-16} refers to the indicated channel.
If the index number is not used, the command applies to channel 1.
- :SOURce{1-4} refers to the indicated external source.
If the index number is not used, the command applies to external source 1.
- :THRu{12 | 13 | 14 | 23 | 24 | 34} refers to the through line between the indicated port pair. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.
If the index number is not used, the command applies to Port 1 and Port 2.
- :TRACe{1-16} refers to the indicated trace on the active channel.
If the index number is not used, the command applies to the trace 1.
- :WINDow refers to the active channel.
- :WINDow{1-16} refers to the indicated channel.
If the index number is not used, the command applies to the window 1.

5-5 :CALCulate{1-16}:AFCW Subsystem

The :CALCulate{1-16}:AFCW (Advanced Fast CW) subsystem provides remote commands to control Advanced Fast CW measurements.

AFCW requires installation of Option 35 and Option 46. AFCW uses the high speed digitizer option to reach maximum acquisition rates of 100 million measurements/sec (adjustable to as slow as < million samples/sec). This mode is always buffered and is accessible from both the user interface and remotely using the commands listed below.

Raw data exporting is available as is pre-processing over variable measurement widths and time ranges.

The maximum buffer size is 50 million measurements in each of four receiver channels (200 million measurements total) without the memory addition of Option 36 and 200 million measurements per receiver (800 million measurements total) *with* Option 36.

:CALCulate{1-16}:AFCW:CDURation <NRf>
:CALCulate{1-16}:AFCW:CDURation?

Description: The command sets the Capture Duration for Advanced Fast CW on the indicated channel.

The query outputs the Capture Duration for Advanced Fast CW on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: NA

Query Output: <NR3> The output is in Seconds.

Range: Dependent on the Sampling Interval of the current channel.

Minimum is 0 and Maximum can range from 500 ms to 14 seconds (without Option 36 – Expanded Digitizer Memory).

Minimum is 0 seconds and Maximum is 2.5 seconds to 70 seconds. (with Option 36 – Expanded Digitizer Memory).

Default Value: 2E-1

Syntax Example: `:CALC1:AFCW:CDUR 1E-1`

`:CALC1:AFCW:CDUR?`

:CALCulate{1-16}:AFCW:DELay{1-8} <NRf>
:CALCulate{1-16}:AFCW:DELay{1-8}?

Description: The command sets the measurement start delay for Advanced Fast CW on an indicated channel and receiver. This applies to both Point-to-Point mode or Full Time Record mode.

The query outputs the measurement start delay for of Advanced Fast CW on an indicated channel and receiver. This applies to both Point-to-Point mode or Full Time Record mode.

Only A1, A2, B1, B2 are allowed receivers for AFCW on a 2 port system.

Receiver definitions:

1 = A1

2 = A2

3 = A3

4 = A4

5 = B1

6 = B2

7 = B3

8 = B4

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: NA

Query Output: <NR3> The output is in Seconds.

Range: 0 s to 2 * Sampling Interval * 10^8 s in 2.5 ns increments

Resolution in pulse modes is equivalent to Sampling Interval in Advanced Fast CW.

Default Value: 0

Syntax Example: `:CALC1:AFCW:DEL1 1E-1`

`:CALC1:AFCW:DEL1?`

:CALCulate{1-16}:AFCW:ESM <char>
:CALCulate{1-16}:AFCW:ESM?

Description: The command sets the External Synch. Marking type for Advanced Fast CW on the given channel.

The query outputs the External Synch. Marking type for Advanced Fast CW on the given channel

Command definitions:

- OFF: External Synch Marking is Off
- RIS: External Rising
- FAL: External Falling

Cmd Parameters: <char> OFF | RIS | FAL

Query Parameters: NA

Query Output: <char> OFF | RIS | FAL

Range: NA

Default Value: OFF

Syntax Example: `:CALC1:AFCW:ESM RIS`

`:CALC1:AFCW:ESM?`

:CALCulate{1-16}:AFCW:ISAM <NRf>
:CALCulate{1-16}:AFCW:ISAM?

Description: The command sets the Sampling Interval for Advanced Fast CW acquisition on the indicated channel.

The query outputs the sampling interval for Advanced Fast CW acquisition on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Seconds.

Range: 2.5E-9 to 70E-9

Default Value: 2.5253E-9

Syntax Example: `:CALC1:AFCW:ISAM 20E-9`

`:CALC1:AFCW:ISAM?`

:CALCulate{1-16}:AFCW:ISYN <NRf>
:CALCulate{1-16}:AFCW:ISYN?

Description: The command sets the Synch. Interval for Advanced Fast CW on the indicated channel.

The query outputs the Synch. Interval for Advanced Fast CW on an indicated channel

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Seconds.

Range: 20 ns to Sampling Interval * 10^8 s in 2.5 ns increments

Resolution in pulse modes is equivalent to Sampling Interval in Advanced Fast CW.

Default Value: 1E-3

Syntax Example: `:CALC1:AFCW:ISYN 4E-5`

`:CALC1:AFCW:ISYN?`

:CALCulate{1-16}:AFCW:MODe <char>
:CALCulate{1-16}:AFCW:MODe?

Description: The command sets the active mode for Advanced Fast CW on the indicated channel.

The query outputs the active mode for Advanced Fast CW on the indicated channel

The active mode definitions are:

- PBP = Point by Point
- FTR = Full Time Record

Cmd Parameters: <char> PBP | FTR

Query Parameters: NA

Query Output: <char> PBP | FTR

Range: NA

Default Value: PBP

Syntax Example: `:CALC1:AFCW:MOD PBP`

`:CALC1:AFCW:MOD?`

:CALCulate{1-16}:AFCW:MWIDth{1-8} <NRf>
:CALCulate{1-16}:AFCW:MWIDth{1-8}?

Description: The command sets the measurement width for Advanced Fast CW on an indicated channel and receiver.

The query outputs the measurement width for Advanced Fast CW on an indicated channel and receiver.

Only A1, A2, B1, B2 are allowed receivers for AFCW on a 2 port system.

Receiver definitions:

1 = A1

2 = A2

3 = A3

4 = A4

5 = B1

6 = B2

7 = B3

8 = B4

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: NA

Query Output: <NR3> The output is in Seconds.

Range: 2.5 ns to 1 * Resolution * 10^8 s in 2.5 ns increments

Max should scale with sampling interval and equal to acquisition length max.

Default Value: 1E-6

Syntax Example: `:CALC1:AFCW:MWID1 2E-6`

`:CALC1:AFCW:MWID1`

:CALCulate{1-16}:AFCW:POINts <NRf>
:CALCulate{1-16}:AFCW:POINts?

Description: The command sets the number of points for Advanced Fast CW on the indicated channel.

The query outputs the number of points for Advanced Fast CW on the indicated channel

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output is an integer.

Range: 1 to 25,000 or 1 to 100,000 (in 100K Mode)

Default Value: 1

Syntax Example: `:CALC1:AFCW:POIN 51`

`:CALC1:AFCW:POIN?`

:CALCulate{1-16}:AFCW:RDEX:BDIRectory <string>
:CALCulate{1-16}:AFCW:RDEX:BDIRectory?

Description: The command sets the target directory for Raw Data Export in binary format for Advanced Fast CW on the indicated channel.

The query outputs target directory for Raw Data Export in binary format for Advanced Fast CW on the indicated.

Cmd Parameters: <string> The input parameter is path to a directory

Query Parameters: NA

Query Output: <string> The output parameter is path to a directory

Range: NA

Default Value: "C:\DJData"

Syntax Example: `:CALCulate{1-16}:AFCW:RDEX:BDIRectory 'C:\AnritsuVNA\Data'`

`:CALCulate{1-16}:AFCW:RDEX:BDIRectory?`

:CALCulate{1-16}:AFCW:RDEX:FORMat <char>
:CALCulate{1-16}:AFCW:RDEX:FORMat?

Description: The command sets the file format for Raw Data Export in Advanced Fast CW on the indicated channel.

The query outputs the file format for Raw Data Export in Advanced Fast CW on the indicated channel

Command definitions:

- OFF = Raw Data Export is off
- BIN = Binary format

Cmd Parameters: <char> OFF|BIN

Query Parameters: NA

Query Output: <char> OFF|BIN

Range: NA

Default Value: OFF

Syntax Example: `:CALC1:AFCW:RDEX:FORM BIN`

`:CALC1:AFCW:RDEX:FORM?`

:CALCulate{1-16}:AFCW:RECeiver:COUPle[:STATe] <char>
:CALCulate{1-16}:AFCW:RECeiver:COUPle[:STATe]?

Description: The command turns receiver coupling on/off for Advanced Fast CW on the given channel.

The query outputs the receiver coupling on/off state for Advanced Fast CW on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Query Output: <char> 1|0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:AFCW:REC:COUP ON`

`:CALC1:AFCW:REC:COUP?`

:CALCulate{1-16}:AFCW:RECeiver:TYPe <char>
:CALCulate{1-16}:AFCW:RECeiver:TYPe?

Description: The command sets the active receiver for Advanced Fast CW on the indicated channel.

The query outputs the active receiver for Advanced Fast CW on the indicated channel

Cmd Parameters: <char> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

Query Parameters: NA

Query Output: <char> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

Range: NA

Default Value: B2

Syntax Example: `:CALC1:AFCW:REC:TYP B1`

`:CALC1:AFCW:REC:TYP?`

:CALCulate{1-16}:AFCW:STOP:DELay{1-8} <NRf>
:CALCulate{1-16}:AFCW:STOP:DELay{1-8}?

Description: The command sets the measurement stop delay for Full Time Record Mode of Advanced Fast CW on an indicated channel and receiver.

The query outputs the measurement stop delay for Full Time Record Mode of Advanced Fast CW on an indicated channel and receiver.

Only A1, A2, B1, B2 are allowed receivers for AFCW on a 2 port system.

Receiver definitions:

1 = A1

2 = A2

3 = A3

4 = A4

5 = B1

6 = B2

7 = B3

8 = B4

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: NA

Query Output: <NR3> The output is in Seconds.

Range: 0 s to 2 * Sampling Interval * 10^8 s in 2.5 ns increments

Resolution in pulse modes is equivalent to Sampling Interval in Advanced Fast CW.

Default Value: 2E-6

Syntax Example: `:CALC1:AFCW:STOP:DEL1 1E-1`

`:CALC1:AFCW:STOP:DEL1?`

5-6 :CALCulate{1-16}:APPLication:MEASurement Subsystem

:CALCulate{1-16}:APPLication:MEASurement:TYPe <char>
:CALCulate{1-16}:APPLication:MEASurement:TYPe?

Description: The command sets the application type on the indicated channel. The query returns the current application type for the indicated channel where the application types are:

- TRANrefl = Transmission/Reflection
- NFIGure = Noise Figure
- PULSe = PulseView™

Changing to Noise Figure Type

When the VNA application mode is changed to Noise Figure type, the instrument configuration changes to the default Noise Figure trace configuration with:

- A single trace
- Noise Figure response in a LogMag display type

Changing to Transmission/Reflection or Pulse Type

When the VNA application mode is changed to Transmission/Reflection or Pulse type, the instrument resets to factory as-shipped default configuration setting of four traces set as:

- S11 response in a Smith Chart display type
- S12 response in a LogMag + Phase display type
- S21 response in a LogMag + Phase display type
- S22 response in a Smith Chart display type

Returning to Another User-Defined Preset Configuration

If a different instrument configuration is required programmatically, use the :SYStem:PRESet command subsystem to apply a previously user-defined configuration. The commands are described in the :SYStem subsystem described below. The available commands are:

```
:SYSTem:PRESet
:SYSTem:PRESet:FILe <string> = 'C:\directory\filename.cha'
:SYSTem:PRESet:FILe?
:SYSTem:PRESet:TYPe <char> = RESET | USER
:SYSTem:PRESet:TYPe?
:SYSTem:PRESet:ZERo
```

Cmd Parameters: <char> TRANrefl | NFIGure | PULSe

Query Parameters: NA

Output: <char>TRAN | NFIG | PULS

Range: NA

Default Value: TRANrefl

Syntax Example: :CALC1:APPL:MEAS:TYP NFIG

:CALC1:APPL:MEAS:TYP?

5-7 :CALCulate{1-16}:CORRection Subsystem - Adapters/Merge Calibration

The :CALCulate{1-16}:CORRection subsystem commands are used to configure and control calibrations related to adapter removal and merge calibration.

Calibration Option Subsystems

Related calibration option configuration and control subsystems are:

- ":CALCulate{1-16}:CORRection Subsystem - Adapters/Merge Calibration" on page 5-14
- ":CALCulate{1-16}:EXTRaction Subsystem - Network Extraction" on page 5-33
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335

:CALCulate{1-16}:CORRection:ADAPter:REMoval

Description: The command performs the adapter removal on the indicated channel. No query.

Cmd Parameters: None

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:CORR:ADAP:REM`

:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:X <string>
:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:X?

Description: The command assigns the calibration X filename to be used in adapter removal on the indicated channel. The query outputs the calibration X filename to be used in adapter removal on the indicated channel. The X filename refers to the calibration done with adapter connected to Port 2.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.chx where the directory and filename must exist.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:CORR:ADAP:REM:CAL:X 'C:\filename.chx'`

`:CALC1:CORR:ADAP:REM:CAL:X?`

:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:Y <string>
:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:Y?

Description: The command assigns the calibration Y filename to be used in adapter removal on the indicated channel. The query outputs the calibration Y filename to be used in adapter removal on the indicated channel. The Y filename refers to the calibration done with adapter connected to Port 1.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.chx

Range: NA

Default Value: NA

Syntax Example: `:CALC1:CORR:ADAP:REM:CAL:Y 'C:\filename.chx'`

`:CALC1:CORR:ADAP:REM:CAL:Y?`

:CALCulate{1-16}:CORRection:ADAPter:REMoval:LENGth <NRf>
:CALCulate{1-16}:CORRection:ADAPter:REMoval:LENGth?

Description: The command inputs the adapter length (in seconds) to be used in adapter removal on the indicated channel. The query outputs the adapter length to be used in adapter removal on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: NA

Query Output: <NR3> The output is in Seconds.

Range: MPND

Default Value: NA

Syntax Example: `:CALC1:CORR:ADAP:REM:LENG 6.6E-11`

`:CALC1:CORR:ADAP:REM:LENG?`

:CALCulate{1-16}:CORRection:MERGe

Description: Performs merge calibration on the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:CORR:MERG`

:CALCulate{1-16}:CORRection:MERGe:CALibration{1-2} <string>
:CALCulate{1-16}:CORRection:MERGe:CALibration{1-2}?

Description: The command assigns the indicated calibration filename to be used in merge calibration on the indicated channel. The query outputs the indicated calibration filename to be used in merge calibration on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.chx

Range: NA

Default Value: NA

Syntax Example: `:CALC1:CORR:MERG:CAL1 'C:\filename.chx'`

`:CALC1:CORR:MERG:CAL1?`

5-8 :CALCulate{1-16}:DISPlay:MARKer Subsystem

The :CALCulate{1-16}:DISPlay:MARKer subsystem command toggles the display of markers on and off.

Marker Subsystems

Related marker configuration, control, and reporting commands are described in the following subsystems:

- ":CALCulate{1-16}:DISPlay:MARKer Subsystem" on page 5-17
- ":CALCulate{1-16}:MARKer Subsystem" on page 5-76
- ":CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem" on page 5-107
- ":CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem" on page 5-110
- ":CALCulate{1-16}[:SELected]:MARKer Subsystem" on page 5-165
- ":CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem" on page 5-178
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}:DISPlay:MARKer:ALL[:STATe] <char>
:CALCulate{1-16}:DISPlay:MARKer:ALL[:STATe]?

Description: Command turns on/off the display of all markers on the given channel. Query outputs the on/off status of displaying the markers on the channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default: 1

Syntax Example: `:CALC1:DISP:MARK:ALL ON`

`:CALC1:DISP:MARK:ALL?`

:CALCulate{1-16}:DISPlay:MARKer:INOVerlay[:STATe] <char>
:CALCulate{1-16}:DISPlay:MARKer:INOVerlay[:STATe]?

Description: Command turns display marker in rectangular overlay mode on/off on the given channel. Query outputs display marker in rectangular overlay mode on/off status on the given channel

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: `:CALC1:DISP:MARK:INOV 1`

`:CALC1:DISP:MARK:INOV?`

5-9 :CALCulate{1-16}:EOOE: Subsystem

:CALCulate{1-16}:EOOE:EO4Measurment:CALCulate?

Description: Deembed the OE Device characterization data from the multiport EO calibration of the given channel and return status.

Cmd Parameters: NA

Query Parameters: <NR1> Query returns one of the following:

0 – Valid

1 – Invalid

2 – InvalidSnPFile

3 – InvalidCHXFile

4 – InvalidCalType

5 – IncompatibleFreq

6 – IncompatiblePort

7 – NoCalExist

8 – WARNING:Extrapolation

9 – InvalidPortSelection

10 – InvalidSnPFileType

11 – InvalidEOFileType

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EOOE:EO4M:CALC?`

:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile <char>
:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile?

Description: Sets the OE device characterization filename for the multiport EO measurement on the given channel. Query outputs the OE device characterization filename for the multiport EO measurement on the given channel.

Cmd Parameters: <char> Path and Filename: "c:\eofiles\myfile1.s2p"

Query Parameters: <char> Query Returns: c:\eofiles\myfile1.s2p

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EOOE:EO4M:CHAR “c:\eofiles\myfile1.s2p”`

`:CALC1:EOOE:EO4M:CHAR?`

:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile:SWAP[:STATe] <char>
:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile:SWAP[:STATe]?

Description: Sets the OE device characterization Swap flag for the multiport EO measurement on the given channel. Query outputs the OE device characterization Swap flag for the multiport EO measurement of the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EOOE:EO4M:CHAR:SWAP ON`

`:CALC1:EOOE:EO4M:CHAR:SWAP?`

:CALCulate{1-16}:EOOE:EO4Measurment:CONFiguration <char>
:CALCulate{1-16}:EOOE:EO4Measurment:CONFiguration?

Description: Sets the configuration of the multiport EO measurement on the given channel. Query outputs the configuration of the multiport EO measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:EOOE:EO4M:CONF PORT12`

`:CALC1:EOOE:EO4M:CONF?`

:CALCulate{1-16}:EOOE:EO4Measurment:EOPort <char>
:CALCulate{1-16}:EOOE:EO4Measurment:EOPort?

Description: Sets the configuration of the multiport EO measurement on the given channel. Query outputs the configuration of the multiport EO measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT2

Syntax Example: `:CALC1:EOOE:EO4M:EOP PORT12`

`:CALC1:EOOE:EO4M:EOP?`

:CALCulate{1-16}:EOOE:EO4Measurment:OEPort <char>
:CALCulate{1-16}:EOOE:EO4Measurment:OEPort?

Description: Sets the OE port assignment for the multiport EO measurement on the given channel. Query outputs the OE port assignment for the multiport EO measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT2

Syntax Example: `:CALC1:EOOE:EO4M:OEP PORT12`

`:CALC1:EOOE:EO4M:OEP?`

:CALCulate{1-16}:EOOE:EOMeasurment:CALCulate?

Description: Deembed the OE Device characterization data from the EO calibration of the given channel and return status. This query performs a Deembedding. In order for it to work, the user must first supply OE device characterization data with an .S2P file and EO calibration data with a .CHX file.

Cmd Parameters: NA

Query Parameters: <NR1> Query returns one of the following:

0 – Valid

1 – Invalid

2 – InvalidSnPFile

3 – InvalidCHXFile

4 – InvalidCalType

5 – IncompatibleFreq

6 – IncompatiblePort

7 – NoCalExist

8 – WARNING:Extrapolation

9 – InvalidPortSelection

10 – InvalidSnPFileType

11 – InvalidEOFileType

Range: NA

Default Value: 3

Syntax Example: `:CALC1:EOOE:EOM:CALC?`

:CALCulate{1-16}:EOOE:EOMeasurment:CALFile <string>
:CALCulate{1-16}:EOOE:EOMeasurment:CALFile?

Description: Sets the calibration filename for the EO measurement on the given channel. Query outputs the calibration filename for the EO measurement of the given channel.

Cmd Parameters: <string> Path and file: "c:\eofiles\myfile1.chx"

Query Parameters: <char> Query returns: c:\eofiles\myfile1.chx

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:EOM:CALF "c:\eofiles\myfile1.chx"`

`:CALC1:EOOE:EOM:CALF?`

:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile <string>
:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile?

Description: Sets the OE device characterization filename for the EO measurement on the given channel. Query outputs the OE device characterization filename for the EO measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.s2p"

Query Parameters: <char> c:\eofiles\myfile1.s2p

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:EOM:CHAR "c:\eofiles\myfile1.s2p"`

`:CALC1:EOOE:EOM:CHAR?`

:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile:SWAP[:STATe] <char>
:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile:SWAP[:STATe]?

Description: Sets the OE device characterization filename for the EO measurement on the given channel. Query outputs the OE device characterization filename for the EO measurement of the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EOOE:EOM:CHAR:SWAP ON`

`:CALC1:EOOE:EOM:CHAR:SWAP?`

:CALCulate{1-16}:EOOE:EOMeasurment:EOPort <char>
:CALCulate{1-16}:EOOE:EOMeasurment:EOPort?

Description: Sets the EO port assignment for the EO measurement on the given channel. Query outputs the EO port assignment for the EO measurement of the given channel.

Cmd Parameters: <char> PORT1 | PORT2

Query Parameters: <char> PORT1 | PORT2

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:EOOE:EOM:EOP PORT1`

`:CALC1:EOOE:EOM:EOP?`

:CALCulate{1-16}:EOOE:GO4Measurment:CALCulate?

Description: Deembed the OE Device characterization data from the EO calibration of the given channel and return status.

Cmd Parameters: NA

Query Parameters: <NR1> Query returns one of the following:

0 – Valid

1 – Invalid

2 – InvalidSnPFile

3 – InvalidCHXFile

4 – InvalidCalType

5 – IncompatibleFreq

6 – IncompatiblePort

7 – NoCalExist

8 – WARNING:Extrapolation

9 – InvalidPortSelection

10 – InvalidSnPFileType

11 – InvalidEOFileType

Range: Any number from the query parameters list

Default Value: 11

Syntax Example: `:CALC1:EOOE:GO4M:CALC?`

:CALCulate{1-16}:EOOE:GO4Measurment:CALFile <string>
:CALCulate{1-16}:EOOE:GO4Measurment:CALFile?

Description: Sets the OE calibration filename for the multiport GO measurement on the given channel. Query outputs the OE calibration filename for the multiport GO measurement of the given channel.

Cmd Parameters: <string>"c:\eofiles\myfile1.chx"

Query Parameters: <char> c:\eofiles\myfile1.chx

Range: N/A

Default: N/A

Syntax Example: `:CALC1:EOOE:GO4M:CALF "c:\eofiles\myfile1.chx"`

`:CALC1:EOOE:GO4M:CALF?`

:CALCulate{1-16}:EOOE:GO4Measurment:CHARfile <char>
:CALCulate{1-16}:EOOE:GO4Measurment:CHARfile?

Description: Sets the OE device characterization filename for the multiport GO measurement on the given channel. Query outputs the OE device characterization filename for the multiport GO measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.s2p"

Query Parameters: <char> c:\eofiles\myfile1.s2p

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:GO4M:CHAR "c:\eofiles\myfile1.s2p"`

`:CALC1:EOOE:GO4M:CHAR?`

:CALCulate{1-16}:EOOE:GO4Measurment:CONFiguration <char>
:CALCulate{1-16}:EOOE:GO4Measurment:CONFiguration?

Description: Sets the OE configuration of the multiport GO measurement on the given channel. Query outputs the OE configuration of the multiport GO measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 |PORT34 |PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 |PORT34 |PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:EOOE:GO4M:CONF PORT12`

`:CALC1:EOOE:GO4M:CONF?`

:CALCulate{1-16}:EOOE:GO4Measurment:EOPort <char>
:CALCulate{1-16}:EOOE:GO4Measurment:EOPort?

Description: Sets the EO port assignment for the multiport GO measurement on the given channel. Query outputs the EO port assignment for the multiport GO measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 |PORT34 |PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 |PORT34 |PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:EOOE:GO4M:EOP PORT1`

`:CALC1:EOOE:GO4M:EOP?`

:CALCulate{1-16}:EOOE:GO4Measurment:OEPort <char>
:CALCulate{1-16}:EOOE:GO4Measurment:OEPort?

Description: Sets the OE port assignment for the multiport GO measurement on the given channel. Query outputs the OE port assignment for the multiport GO measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 |PORT34 |PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 |PORT34 |PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT2

Syntax Example: `:CALC1:EOOE:GO4M:EOP Port12`

`:CALC1:EOOE:GO4M:EOP?`

:CALCulate{1-16}:EOOE:GO4Measurment:TARGetfile <string>
:CALCulate{1-16}:EOOE:GO4Measurment:TARGetfile?

Description: Sets the EO device characterization target filename to use for the multiport GO measurement on the given channel. Query outputs the EO device characterization target filename to use for the multiport GO measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.s2p"

Query Parameters: <char> c:\eofiles\myfile1.s2p

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:GO4M:TARG "c:\eofiles\myfile1.s2p"`

`:CALC1:EOOE:GO4M:TARG?`

:CALCulate{1-16}:EOOE:GOMeasurment:CALCulate?

Description: Measure and store the EO device characterization data to the target file of the given channel and return status.

Cmd Parameters: NA

Query Parameters: <NR3>

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EOOE:GOM:CALC?`

:CALCulate{1-16}:EOOE:GOMeasurment:CALFile <string>
:CALCulate{1-16}:EOOE:GOMeasurment:CALFile?

Description: Sets the calibration filename for the GO measurement on the given channel. Query outputs the calibration filename for the GO measurement of the given channel.

Cmd Parameters: <string>"c:\eofiles\myfile1.chx"

Query Parameters: <char> c:\eofiles\myfile1.chx

Range: N/A

Default: N/A

Syntax Example: `:CALC1:EOOE:GOM:CALF "c:\eofiles\myfile1.chx"`

`:CALC1:EOOE:GOM:CALF?`

:CALCulate{1-16}:EOOE:GOMeasurment:CHARfile <char>
:CALCulate{1-16}:EOOE:GOMeasurment:CHARfile?

Description: Sets the OE device characterization filename for the GO measurement on the given channel. Query outputs the OE device characterization filename for the GO measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.s2p"

Query Parameters: <char> c:\eofiles\myfile1.s2p

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:GO4M:CHAR "c:\eofiles\myfile1.s2p"`

`:CALC1:EOOE:GO4M:CHAR?`

:CALCulate{1-16}:EOOE:GOMeasurment:EOPort <char>
:CALCulate{1-16}:EOOE:GOMeasurment:EOPort?

Description: Sets the EO port assignment for the GO measurement on the given channel. Query outputs the EO port assignment for the GO measurement of the given channel.

Cmd Parameters: <char> PORT1|PORT2

Query Parameters: <char> PORT1|PORT2

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:EOOE:GOM:EOP PORT1`

`:CALC1:EOOE:GOM:EOP?`

:CALCulate{1-16}:EOOE:GOMeasurment:TARGetfile <string>
:CALCulate{1-16}:EOOE:GOMeasurment:TARGetfile?

Description: Sets the EO device characterization target filename to use for the GO measurement on the given channel. Query outputs the EO device characterization target filename to use for the GO measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.s2p"

Query Parameters: <char> c:\eofiles\myfile1.s2p

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:GOM:TARG "c:\eofiles\myfile1.s2p"`

`:CALC1:EOOE:GOM:TARG?`

:CALCulate{1-16}:EOOE:MSGS:LIST?

Description: Outputs a copy of the EOOE messages list.

Cmd Parameters: NA

Query Parameters: <arbitrary block data>

Range: N/A

Default Value: NA

Syntax Example: `:CALC1:EOOE:MSGS:LIST?`

```
Query returns the following list:
0 – Valid
1 – Invalid
2 – InvalidSnPFile
3 – InvalidCHXFile
4 – InvalidCalType
5 – IncompatibleFreq
6 – IncompatiblePort
7 – NoCalExist
8 – WARNING:Extrapolation
9 – InvalidPortSelection
10 – InvalidSnPFileType
11 – InvalidEOFileType
```

:CALCulate{1-16}:EOOE:OE4Measurment:CALCulate?

Description: Deembed the EO Device characterization data from the multiport OE calibration of the given channel and return status.

Cmd Parameters: NA

Query Parameters: <NR1> Query returns one of the following:

0 – Valid

1 – Invalid

2 – InvalidSnPFile

3 – InvalidCHXFile

4 – InvalidCalType

5 – IncompatibleFreq

6 – IncompatiblePort

7 – NoCalExist

8 – WARNING:Extrapolation

9 – InvalidPortSelection

10 – InvalidSnPFileType

11 – InvalidEOFileType

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EOOE:OE4M:CALC? 3`

:CALCulate{1-16}:EOOE:OE4Measurment:CALFile <string>
:CALCulate{1-16}:EOOE:OE4Measurment:CALFile?

Description: Sets the calibration filename for the multiport OE measurement on the given channel. Query outputs the calibration filename for the multiport OE measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.chx"

Query Parameters: <char> c:\eofiles\myfile1.chx

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:OE4M:CALF "c:\eofiles\myfile1.chx"`

`:CALC1:EOOE:OE4M:CALF?`

:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile <string>
:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile?

Description: Sets the EO device characterization filename for the multiport OE measurement on the given channel. Query outputs the EO device characterization filename for the multiport OE measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.s2p"

Query Parameters: <char> c:\eofiles\myfile1.s2p

Range: N/A

Default Value: N/A

Syntax Example: `:CALC1:EOOE:OE4M:CHAR "c:\eofiles\myfile1.s2p"`

`:CALC1:EOOE:OE4M:CHAR?`

:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile:SWAP[:STATe] <char>
:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile:SWAP[:STATe]?

Description: Sets the EO device characterization Swap flag for the multiport OE measurement on the given channel. Query outputs the EO device characterization Swap flag for the multiport OE measurement of the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0|

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EOOE:OE4M:CHAR:SWAP ON`

`:CALC1:EOOE:OE4M:CHAR:SWAP?`

:CALCulate{1-16}:EOOE:OE4Measurment:CONFiguration <char>
:CALCulate{1-16}:EOOE:OE4Measurment:CONFiguration?

Description: Sets the configuration of the multiport OE measurement on the given channel. Query outputs the configuration of the multiport OE measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT2

Syntax Example: `:CALC1:EOOE:OE4M:CONF PORT12`

`:CALC1:EOOE:OE4M:CONF?`

:CALCulate{1-16}:EOOE:OE4Measurment:EOPort <char>
:CALCulate{1-16}:EOOE:OE4Measurment:EOPort?

Description: Sets the EO port assignment for the multiport OE measurement on the given channel. Query outputs the EO port assignment for the multiport OE measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:EOOE:OE4M:EOP PORT12`

`:CALC1:EOOE:OE4M:EOP?`

:CALCulate{1-16}:EOOE:OE4Measurment:OEPort <char>
:CALCulate{1-16}:EOOE:OE4Measurment:OEPort?

Description: Sets the OE port assignment for the multiport OE measurement on the given channel. Query outputs the OE port assignment for the multiport OE measurement of the given channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT2

Syntax Example: `:CALC1:EOOE:OE4M:OEP PORT12`

`:CALC1:EOOE:OE4M:OEP?`

:CALCulate{1-16}:EOOE:OEMeasurment:CALCulate?

Description: Deembed the EO Device characterization data from the OE calibration of the given channel and return status.

Cmd Parameters: NA

Query Parameters: <NR1> Query returns one of the following:

0 – Valid

1 – Invalid

2 – InvalidSnPFile

3 – InvalidCHXFile

4 – InvalidCalType

5 – IncompatibleFreq

6 – IncompatiblePort

7 – NoCalExist

8 – WARNING:Extrapolation

9 – InvalidPortSelection

10 – InvalidSnPFileType

11 – InvalidEOFileType

Range: NA

Default Value: 3

Syntax Example: `:CALC1:EOOE:OEM:CALC? 3`

:CALCulate{1-16}:EOOE:OEMeasurment:CALFile <string>
:CALCulate{1-16}:EOOE:OEMeasurment:CALFile?

Description: Sets the calibration filename for the OE measurement on the given channel. Query outputs the calibration filename for the OE measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.chx"

Query Parameters: <char> c:\eofiles\myfile1.chx

Range: N/A

Default: N/A

Syntax Example: `:CALC1:EOOE:OEM:CALF "c:\eofiles\myfile1.chx"`

`:CALC1:EOOE:OEM:CALF?`

:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile <string>
:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile?

Description: Sets the EO device characterization filename to use for the OE measurement on the given channel. Query outputs the EO device charaterization filename to use for the OE measurement of the given channel.

Cmd Parameters: <string> "c:\eofiles\myfile1.s2p"

Query Parameters: <char> c:\eofiles\myfile1.s2p

Range: N/A

Default: N/A

Syntax Example: `:CALC1:EOOE:OEM:CHAR "c:\eofiles\myfile1.s2p"`

`:CALC1:EOOE:OEM:CHAR?`

:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile:SWAP[:STATe] <char>
:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile:SWAP[:STATe]?

Description: Sets the EO device characterization Swap flag for the OE measurement on the given channel. Query outputs the EO device characterization Swap flag for the OE measurement on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EOOE:OEM:CHAR:SWAP ON`

`:CALC1:EOOE:OEM:CHAR:SWAP?`

:CALCulate{1-16}:EOOE:OEMeasurment:EOPort <char>
:CALCulate{1-16}:EOOE:OEMeasurment:EOPort?

Description: Sets the EO port assignment for the OE measurement on the given channel. Query outputs the EO port assignment for the OE measurement on the given channel.

Cmd Parameters: <char> PORT1|PORT2

Query Parameters: <char> PORT1|PORT2

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:EOOE:OEM:EOP PORT1`

`:CALC1:EOOE:OEM:EOP?`

:CALCulate{1-16}:EOOE:TYPE?

Description: Query only. Outputs the type of EOOE measurement that is currently loaded on the instrument.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <char> EO | OE | OO | NONE

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EOOE:TYPE?`

5-10 :CALCulate{1-16}:EXTRaction Subsystem - Network Extraction

The :CALCulate{1-16}:EXTRaction subsystem commands provide configuration control and execution for network extraction functions during an instrument calibration.

Calibration Option Subsystems

Related calibration option configuration and control subsystems are:

- “:CALCulate{1-16}:CORRection Subsystem - Adapters/Merge Calibration” on page 5-14
- “:CALCulate{1-16}:EXTRaction Subsystem - Network Extraction” on page 5-33
- “:CALCulate{1-16}:NXN Subsystem” on page 5-88
- “:SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem” on page 5-288
- “:SENSe{1-16}:CORRection:COLLect:HYBRid Subsystem” on page 5-290

General Parameters

The general command parameters are:

- [:METHod]:A refers to Extraction Method Type A which extracts one 2-port network using the adapter extraction method. Available on 2-Port and 4-Port VNA instruments.
- [:METHod]:B refers to Extraction Method Type B which extracts one 2-port network using a two-tier calibration. Available on 2-Port and 4-Port VNA instruments.
- [:METHod]:C refers to Extraction Method Type C which extracts two 2-port networks using inner and outer calibrations. Available on 2-Port and 4-Port VNA instruments.
- [:METHod]:D refers to Extraction Method Type D which extracts two Two-Port networks using outer calibrations only using the divide-by-two method. Available on 2-Port and 4-Port VNA instruments.
- [:METHod]:E refers to Extraction Method Type E which extracts four Two-Port networks using inner and outer calibrations. Only available on 4-Port VNA instruments.
- [:METHod]:F refers to Extraction Method Type F which extracts four Two-Port networks with outer calibrations only using the divide-by-two method. Only available on 4-Port VNA instruments.
- [:METHod]:G refers to Extraction Method Type G which extracts two Four-Port networks with outer calibrations only using the divide-by-two method. Only available on 4-Port VNA instruments.

:CALCulate{1-16}:EXTRaction

Description: The command performs the network extraction after using the network extraction setup commands below. This is the same as :CALCulate{1-16}:EXTRaction[:METHod]:C described below.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EXTR`

:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:FILe <string>
:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:FILe?

Description: Assigns the Calibration A filename to be used in Network Extraction on the indicated channel.

Returns the Calibration A filename that is to be used in Network Extraction on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.chx

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EXTR:CAL:CAL 'C:\directory\cala.chx'`

`:CALC1:EXTR:CAL:CAL?`

:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:PORT <char>
:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:PORT?

Description: Assigns the Calibration A port to be used in Network Extraction on the indicated channel. The use of Port 3 or Port 4 requires a 4-Port VNA instrument.

Returns the Calibration A Port to be used in the Network Extraction on the indicated channel.

Cmd Parameters: <char> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: NA

Query Output: <char> PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EXTR:CAL:CAL:PORT PORT3`

`:CALC1:EXTR:CAL:CAL:PORT?`

:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe <string>
:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe?

Description: This command requires a 4-Port VNA instrument. Assigns the Calibration B filename to be used in Network Extraction on the indicated channel.

Returns the Calibration B filename to be used in Network Extraction on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.chx

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EXTR:CAL:CALB 'C:\directory\filename.chx'`

`:CALC1:EXTR:CAL:CALB?`

:CALCulate{1-16}:EXTRaction:CALibration:CALB:PORT <char>
:CALCulate{1-16}:EXTRaction:CALibration:CALB:PORT?

Description: This command requires a 4-Port VNA instrument. Assigns the Calibration B port to be used in Network Extraction on the indicated channel.

Returns the Calibration B port to be used in the Network Extraction on the indicated channel.

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: NA

Query Output: <char> PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EXTR:CAL:CALB:PORT PORT4`

`:CALC1:EXTR:CAL:CALB:PORT?`

:CALCulate{1-16}:EXTRaction:CALibration:INNer <string>
:CALCulate{1-16}:EXTRaction:CALibration:INNer?

Description: The command assigns the inner calibration filename to be used in network extraction on the indicated channel. The query outputs the inner calibration filename to be used in network extraction on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.chx

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EXTR:CAL:CAL:INN 'C:\filename.chx'`

`:CALC1:EXTR:CAL:CAL:INN?`

:CALCulate{1-16}:EXTRaction:CALibration:OUTer <string>
:CALCulate{1-16}:EXTRaction:CALibration:OUTer?

Description: The command assigns the outer calibration filename to be used in network extraction on the indicated channel. The query outputs the outer calibration filename to be used in network extraction on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'X:\directory\filename.chx' where x:\directory\filename.chx must exist.

Query Parameters: <char> Filename and path in the form: X:\directory\filename.chx

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EXTR:CAL:OUT 'C:\directory\filename.chx'`

`:CALC1:EXTR:CAL:OUT?`

:CALCulate{1-16}:EXTRaction:ELL1:LENGth <NRf>
:CALCulate{1-16}:EXTRaction:ELL1:LENGth?

Description: Sets the electric length 1 of the given network to be used in Network Extraction on the indicated channel. The query outputs the length 1 of the given network to be used in Network Extraction on the indicated channel.

The ELL1 value is used in the following extraction methods:

- Type A = Extract one 2-port network, with adapter extraction
- Type B = Extract one 2-port network, with two tier calibration
- Type D = Extract two 2-port networks, with outer calibration only, using divide-by-two method
- Type E = Extract four 2-port networks, with inner and outer calibrations available
- Type F = Extract four 2-port networks, with outer calibration only, using divide-by-two method
- Type G = Extract two 4-port networks, with outer calibration only, using divide-by-two method

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Meters.

Range: NA

Default: 0

Syntax Example: `:CALC1:EXTR:ELL1:LENG 2.5E-10`

`:CALC1:EXTR:ELL1:LENG?`

:CALCulate{1-16}:EXTRaction:ELL2:LENGth <NRf>
:CALCulate{1-16}:EXTRaction:ELL2:LENGth?

Description: Sets the electric length 2 of the given network to be used in Network Extraction on the indicated channel. The query outputs the length 2 of the given network to be used in Network Extraction on the indicated channel.

The ELL2 value is used in the following extraction methods:

- Type A - Extract one 2-port network, with adapter extraction
- Type E - Extract four 2-port networks, with inner and outer calibrations available
- Type F - Extract four 2-port networks, with outer calibration only, using divide-by-two method

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0

Syntax Example: `:CALC1:EXTR:ELL2:LENG 2.5E-10`

`:CALC1:EXTR:ELL2:LENG?`

:CALCulate{1-16}:EXTRaction:ELL3:LENGth <NRf>
:CALCulate{1-16}:EXTRaction:ELL3:LENGth?

Description: Sets the electric length 3 of the given network to be used in Network Extraction on the indicated channel. The query outputs the length 3 of the given network to be used in Network Extraction on the indicated channel description.

The ELL3 value is used in the following extraction methods:

- Type E - Extract four 2-port networks, with inner and outer calibrations available

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0

Syntax Example: `:CALC1:EXTR:ELL3:LENG 2.5E-10`

`:CALC1:EXTR:ELL3:LENG?`

:CALCulate{1-16}:EXTRaction:ELL4:LENGth <NRf>
:CALCulate{1-16}:EXTRaction:ELL4:LENGth?

Description: Sets the electric length 4 of the given Network to be used in Network Extraction on the indicated channel. The query outputs the length 4 of the given Network to be used in Network Extraction on the indicated channel description.

The ELL4 value is used in the following extraction methods:

- Type E - Extract four 2-port networks, with inner and outer calibrations available

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: NA

Query Output: <NR3> The input parameter is in Meters.

Range: MPND

Default: 0

Syntax Example: `:CALC1:EXTR:ELL4:LENG`

`:CALC1:EXTR:ELL4:LENG?`

:CALCulate{1-16}:EXTRaction:S2P1filename:FILe <string>
:CALCulate{1-16}:EXTRaction:S2P1filename:FILe?

Description: Assigns the S2P file 1 name which receives the Extracted Network S2P data for the indicated port on the indicated channel. The query outputs the S2P file 1 name which receives the Extracted Network S2P data for the indicated port on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p.

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:S2P1:FIL 'C:\directory\filename.s2p'`

`:CALC1:EXTR:S2P1:FIL?`

:CALCulate{1-16}:EXTRaction:S2P2filename:FILe <string>
:CALCulate{1-16}:EXTRaction:S2P2filename:FILe?

Description: Assigns the S2P file 2 name which receives the Extracted Network S2P data for the indicated port on the indicated channel. The query outputs the S2P file 2 name which receives the Extracted Network S2P data for the indicated port on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p.

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:S2P2:FIL 'C:\directory\filename.s2p'`

`:CALC1:EXTR:S2P2:FIL?`

:CALCulate{1-16}:EXTRaction:S2P3filename:FILe <string>
:CALCulate{1-16}:EXTRaction:S2P3filename:FILe?

Description: Assigns the S2P file 3 name which receives the Extracted Network S2P data for the indicated port on the indicated channel. The query outputs the S2P file 3 name which receives the Extracted Network S2P data for the indicated port on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p.

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:S2P3:FIL 'C:\directory\filename.s2p'`

`:CALC1:EXTR:S2P3:FIL?`

:CALCulate{1-16}:EXTRaction:S2P4filename:FILe <string>
:CALCulate{1-16}:EXTRaction:S2P4filename:FILe?

Description: Assigns the S2P file 4 name which receives the Extracted Network S2P data for the indicated port on the indicated channel. The query outputs the S2P file 4 name which receives the Extracted Network S2P data for the indicated port on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p.

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:S2P4:FIL 'C:\directory\filename.s2p'`

`:CALC1:EXTR:S2P4:FIL?`

:CALCulate{1-16}:EXTRaction:S4P1filename:FILe <string>
:CALCulate{1-16}:EXTRaction:S4P1filename:FILe?

Description: Assigns the S4P file 1 name which receives the Extracted Network S4P data for the indicated port on the indicated channel. The query outputs the S4P file 1 name which receives the Extracted Network S4P data for the indicated port on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s4p' where x:\directory\ must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s4p.

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:S4P1:FIL 'C:\directory\filename.s4p'`

`:CALC1:EXTR:S4P1:FIL?`

:CALCulate{1-16}:EXTRaction:S4P2filename:FILe <string>
:CALCulate{1-16}:EXTRaction:S4P2filename:FILe?

Description: Assigns the S4P file 2 name which receives the Extracted Network S4P data for the indicated port on the indicated channel. The query outputs the S4P file 2 name which receives the Extracted Network S4P data for the indicated port on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s4p' where x:\directory\ must exist.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s4p.

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:S4P2:FIL 'C:\directory\filename.s4p'`

`:CALC1:EXTR:S4P2:FIL?`

:CALCulate{1-16}:EXTRaction:SXPPortpair:PORT <char>
:CALCulate{1-16}:EXTRaction:SXPPortpair:PORT?

Description: Assigns the data port set to use when creating an S2P or S4P data file on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the data port set assigned to use when creating an S2P or S4P data file on the indicated.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Query Parameters: NA

Query Output: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:SXPP:PORT23`

`:CALC1:EXTR:SXPP:PORT?`

:CALCulate{1-16}:EXTRaction:ZERo:MATCh[:STATe] <char>
:CALCulate{1-16}:EXTRaction:ZERo:MATCh[:STATe]?

Description: Sets the true/false state of the zero match terms flag on the indicated channel. Outputs the true/false state of the zero match terms flag on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EXTR:ZER:MATC 1`

`:CALC1:EXTR:ZER:MATC?`

:CALCulate{1-16}:EXTRaction[:METHod]:A

Description: Performs the Network Extraction using Method A on the given channel. Method Type A extracts one 2-port network using the adapter extraction method. Available on 2-Port and 4-Port VNA instruments.

No query.

The following commands must be sent before performing network extraction Method A:

- :CALCulate{1-16}:EXTRaction:CALibration:CALa:FILe
- `:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe`
- `:CALCulate{1-16}:EXTRaction:CALa:PORT`
- `:CALCulate{1-16}:EXTRaction:CALB:PORT`
- `:CALCulate{1-16}:EXTRaction:ELL1:LENGth`
- `:CALCulate{1-16}:EXTRaction:S2P1filename:FILe`

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:METH:A`

:CALCulate{1-16}:EXTRaction[:METHod]:B

Description: Performs the Network Extraction using Method B on the given channel. Method Type B extracts one 2-port network using a two-tier calibration. Available on 2-Port and 4-Port VNA instruments.

No query.

The following commands must be sent before performing network extraction Method B:

- `:CALCulate{1-16}:EXTRaction:CALibration:CALa:FILe`
- `:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe`
- `:CALCulate{1-16}:EXTRaction:CALibration:CALa:PORT`
- `:CALCulate{1-16}:EXTRaction:ELL1:LENGth`
- `:CALCulate{1-16}:EXTRaction:S2P1filename:FILe`

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:METH:B`

:CALCulate{1-16}:EXTRaction[:METHod]:C

Description: Performs the Network Extraction using Method C on the given channel. Method Type C extracts two 2-port networks using inner and outer calibrations. Available on 2-Port and 4-Port VNA instruments.

No query.

The following commands must be sent before performing network extraction Method C:

- `:CALCulate{1-16}:EXTRaction:CALibration:CALa:FILe`
- `:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe`
- `:CALCulate{1-16}:EXTRaction:SXPPortpair:PORT`
- `:CALCulate{1-16}:EXTRaction:S2P1filename:FILe`
- `:CALCulate{1-16}:EXTRaction:S2P2filename:FILe`

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:METH:C`

:CALCulate{1-16}:EXTRaction[:METHod]:D

Description: Performs the Network Extraction using Method D on the given channel. Method Type D extracts two Two-Port networks using outer calibrations only using the divide-by-two method. Available on 2-Port and 4-Port VNA instruments.

No query.

The following commands must be sent before performing network extraction Method D:

- `:CALCulate{1-16}:EXTRaction:ZERO:MATCh[:STATe]`
- `:CALCulate{1-16}:EXTRaction:ELL1:LENGth`
- `:CALCulate{1-16}:EXTRaction:SXPPortpair:PORT`
- `:CALCulate{1-16}:EXTRaction:S2P1filename:FILe`
- `:CALCulate{1-16}:EXTRaction:S2P2filename:FILe`

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:METH:D`

:CALCulate{1-16}:EXTRaction[:METHod]:E

Description: Performs the Network Extraction using Method E on the given channel. Method Type E extracts four Two-Port networks using inner and outer calibrations. Only available on 4-Port VNA instruments.

No query.

The following commands must be sent before performing network extraction Method E:

- `:CALCulate{1-16}:EXTRaction:CALibration:CALa:FILe`
- `:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe`
- `:CALCulate{1-16}:EXTRaction:ELL1:LENGth`
- `:CALCulate{1-16}:EXTRaction:ELL2:LENGth`
- `:CALCulate{1-16}:EXTRaction:ELL3:LENGth`
- `:CALCulate{1-16}:EXTRaction:ELL4:LENGth`
- `:CALCulate{1-16}:EXTRaction:S2P1filename:FILe`
- `:CALCulate{1-16}:EXTRaction:S2P2filename:FILe`
- `:CALCulate{1-16}:EXTRaction:S2P3filename:FILe`
- `:CALCulate{1-16}:EXTRaction:S2P4filename:FILe`

Cmd Parameters: NA

Query Parameters: NA

Query Outputs: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:METH:E`

:CALCulate{1-16}:EXTRaction[:METHod]:F

Description: Performs the Network Extraction using Method F on the given channel. Method Type F extracts four Two-Port networks with outer calibrations only using the divide-by-two method. Only available on 4-Port VNA instruments. No query.

The following commands must be sent before performing network extraction Method F:

- :CALCulate{1-16}:EXTRaction:ZERO:MATCh[:STATe]
- :CALCulate{1-16}:EXTRaction:ELL1:LENGth
- :CALCulate{1-16}:EXTRaction:ELL2:LENGth
- :CALCulate{1-16}:EXTRaction:S2P1filename:FILe
- :CALCulate{1-16}:EXTRaction:S2P2filename:FILe
- :CALCulate{1-16}:EXTRaction:S2P3filename:FILe
- :CALCulate{1-16}:EXTRaction:S2P4filename:FILe

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:METH:F`

:CALCulate{1-16}:EXTRaction[:METHod]:G

Description: Performs the Network Extraction using Method G on the given channel. Method Type G extracts two four-port networks with outer calibrations only using the divide-by-two method. Only available on 4-Port VNA instruments.

No query.

The following commands must be sent before performing network extraction Method G:

- :CALCulate{1-16}:EXTRaction:ZERO:MATCh[:STATe]
- :CALCulate{1-16}:EXTRaction:ELL1:LENGth
- :CALCulate{1-16}:EXTRaction:SXPPortpair:PORT
- :CALCulate{1-16}:EXTRaction:S4P1filename:FILe
- :CALCulate{1-16}:EXTRaction:S4P2filename:FILe

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:EXTR:METH:G`

5-11 :CALCulate{1-16}:FCW Subsystem

The :CALCulate{1-16}:FCW subsystem provides remote commands to control Standard Fast CW measurements.

FCW uses the standard MS464XX IF system which allows maximum acquisition rates of ~200,000 measurements/sec (and as slow as 1 measurement/sec). This mode is only available remotely to maximize speed and the display does not update in this mode.

There are two main sub-modes: real-time streaming over the remote interface with unlimited data collection capabilities, and a buffered collection with a maximum buffer size of ~100 million measurements (dependent on available memory) for a single S-parameter.

:CALCulate{1-16}:FCW:CPCount?

Description: Query only. Returns the count of the number of points that has been collected so far in the internal buffer for the selected channel

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1> The number of data points captured thus far

Range: 0 – (Maximum storage)

Default: 0

Syntax Example: `:CALC1:FCW:CPC?`

:CALCulate{1-16}:FCW:DATA?

Description: Query Only. The query outputs the data collected thus far in the form set by the current mode.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <arbitrary block data> formatted with the real and imaginary values as 4 byte floating point numbers:

Header example: #9000000008. The header starts with an octothorpe or pound sign, #. Then follows a number telling how many more digits to follow, in this case 9, then the number of bytes that are to be read, in this case 8.

S-Parameter mode: The header, then sets of 2 floating point numbers, with each number being 4 bytes long.

Receiver mode: The header, then sets of 6 floating point numbers (two for each receiver), with each number being 4 bytes long.

Range: NA

Default: NA

Syntax Example: `:CALC1:FCW:DATA?`

:CALCulate{1-16}:FCW:DCOLlect <char>
:CALCulate{1-16}:FCW:DCOLlect?

Description: This command sets whether the system should start, stop, or pause (hold) collecting Fast CW data.

The query outputs the current data collection setting.

Note that this setting must be set to CONT for data to be collected.

Setting Definitions:

- CONT = Start or continue collecting data
- HOLD = Pause collecting data
- STOP = Stop collecting data
- STREAM = Constantly collect and transmit data

Cmd Parameters: <char> CONT | HOLD | STOP | STREAM

Query Parameters: NA

Query Output: <char> CONT | HOLD | STOP | STREAM

Range: NA

Default: STOP

Syntax Example: `:CALC1:FCW:DCOL CONT`

`:CALC1:FCW:DCOL?`

:CALCulate{1-16}:FCW:DTYPe <char>
:CALCulate{1-16}:FCW:DTYPe?

Description: The command sets the type of data to be collected in the internal buffer.

The query outputs the type of data that will be collected

Definitions:

- CORR = Corrected data
- FIN = Final data
- RAW = Raw data

Cmd Parameters: <char> CORRected | FINal | RAW

Query Parameters: NA

Query Output: <char> CORR | FIN | RAW

Range: NA

Default: FIN

Syntax Example: `:CALC1:FCW:DTYP RAW`

`:CALC1:FCW:DTYP?`

:CALCulate{1-16}:FCW:IBUF:POINts <NRf>
:CALCulate{1-16}:FCW:IBUF:POINts?

Description: The command sets the selected number of points for internal buffer mode of FastCW in the given channel.

The query outputs the selected number of points for internal buffer mode of FastCW in the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: Limited by system memory

Default: 1,000,000

Syntax Example: `:CALC1:FCW:IBUF:POIN 1000000`

`:CALC1:FCW:IBUF:POIN?`

:CALCulate{1-16}:FCW:MARK <char>

Description: This command inserts a marker in the collected data to allow for periodic measurements to be identified. The parameter is inserted as an 8 byte element in the real part of the nearest available data point position and the imaginary part of that point is filled with a 0.

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:FCW:MARK`

:CALCulate{1-16}:FCW:MODE <char>
:CALCulate{1-16}:FCW:MODE?

Description: The command sets whether Fast CW will return the S-parameter data, or the receiver data.

SPAR returns the complex data corresponding the response parameter in the first trace (S-parameter or user defined).

The RCVR mode will return 3 complex variables per point (both test receivers and the reference receiver corresponding to the driving port of the response parameter in the first trace).

The query returns the current Fast CW mode.

Definitions:

- SPAR = S-parameter data
- RCVR = receiver data

Cmd Parameters: <char> SPARameter | RCVR

Query Parameters: NA

Query Output: <char> SPAR | RCVR

Range: NA

Default: SPAR

Syntax Example: `:CALC1:FCW:MODE RCVR`

`:CALC1:FCW:MODE?`

:CALCulate{1-16}:FCW:MPCount?

Description: Query only. Returns the maximum amount of points that can be collected in Fast CW mode by this channel.

Query Output: <NR1> The maximum number of points that can be collected on this channel

Range: Limited by system memory

Default: 1,000,000

Syntax Example: `:CALC1:FCW:MPC?`

:CALCulate{1-16}:FCW[:STATe] <char>
:CALCulate{1-16}:FCW[:STATe]?

Description: The command enables Fast CW mode on the selected channel.

The query returns the state of Fast CW mode on the selected channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: `:CALC:FCW 1`

`:CALC:FCW?`

:CALCulate{1-16}:FCW:STReam:POINts <NRf>
:CALCulate{1-16}:FCW:STReam:POINts?

Description: The command sets the selected number of points for streaming mode of FastCW in the given channel.

The query outputs the selected number of points for streaming mode of FastCW in the given channel

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: 1 to 500

Default: 1 content

Syntax Example: `:CALC1:FCW:STR:POIN 51`

`:CALC1:FCW:STR:POIN?`

5-12 :CALCulate{1-16}:FORMat Subsystem - SnP Data

The :CALCulate{1-16}:FORMat subsystem commands assign data ports when creating SnP data files. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- “:CALCulate{1-16}:FORMat Subsystem - SnP Data” on page 5-49
- “:CALCulate{1-16}:NXN Subsystem” on page 5-88
- “:CALCulate{1-16}[:SELected]:DATA Subsystem” on page 5-136
- “:CALCulate{1-16}[:SELected]:TDATA Subsystem” on page 5-197
- “:FORMat Subsystem” on page 5-242
- “:HCOPy Subsystem” on page 5-245
- “:MMEMory Subsystem” on page 5-251

:CALCulate{1-16}:FORMat:S1P:PORT <char>
:CALCulate{1-16}:FORMat:S1P:PORT?

Description: The command assigns the data port to use when creating an S1P data file on the indicated channel. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. The query outputs the data port assigned to use when creating an S1P data file on the indicated channel.

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: NA

Query Output: NA

Query Output: <char> PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default: PORT1

Syntax Example: `:CALC1:FORM:S1P:PORT PORT1`

`:CALC1:FORM:S1P:PORT?`

:CALCulate{1-16}:FORMat:S2P:PORT <char>
:CALCulate{1-16}:FORMat:S2P:PORT?

Description: The command assigns the data port pair to use when creating an S2P data file on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the data port pair assigned to use when creating an S2P data file on the indicated channel.

Cmd Parameters: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Query Parameters: NA

Query Output: <char> PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Range: NA

Default: PORT12

Syntax Example: `:CALC1:FORM:S2P:PORT PORT12`

`:CALC1:FORM:S2P:PORT?`

:CALCulate{1-16}:FORMat:S3P:PORT <char>
:CALCulate{1-16}:FORMat:S3P:PORT?

Description: This command requires a 4-Port VNA instrument. The command assigns the data port triplet to use when creating an S3P data file on the indicated channel. The query outputs the data port triplet assigned to use when creating an S3P data file on the indicated channel.

Cmd Parameters: <char> PORT123 | PORT124 | PORT134 | PORT234

Query Parameters: NA

Query Output: <char> PORT123 | PORT124 | PORT134 | PORT234

Range: NA

Default Value: PORT123

Syntax Example: :CALC1:FORM:S3P:PORT PORT123

:CALC1:FORM:S3P:PORT?

5-13 :CALCulate{1-16}:FSIMulator:NETWork Subsystem - Simulation

The :CALCulate{1-16}:FSIMulator:NETWork subsystem commands use existing calibration files with a simulated network of various types to evaluate predicted performance. The commands apply to the active network.

Calibration Simulation Subsystems

These subsystems are used to create a calibrated state in the instrument which is followed by adding the required error correction coefficients for the required calibration type. If this approach is used, each error correction coefficient is entered by separate commands. Simulated calibration subsystems are:

- ":CALCulate{1-16}:FSIMulator:NETWork Subsystem - Simulation" on page 5-51
- ":CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation" on page 5-63
- ":SENSe{1-16}:CORRection:COEFficient:PORT Subsystem - Simulation" on page 5-269
- ":SENSe{1-16}:CORRection:COEFficient Subsystem - Simulation" on page 5-274

:CALCulate{1-16}:FSIMulator:NETWork:ADD

Description: The command adds a blank network to be defined on the indicated channel. No query.

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:FSIM:NETW:ADD`

:CALCulate{1-16}:FSIMulator:NETWork:C <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:C?

Description: The command sets the current LC network capacitance value on the indicated channel. The query outputs the current LC network capacitance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Farads.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:C 3.0E-12`

`:CALC1:FSIM:NETW:C?`

:CALCulate{1-16}:FSIMulator:NETWork:CLEar

Description: The command clears all networks on the indicated channel. No query.

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:FSIM:NETW:CLE`

:CALCulate{1-16}:FSIMulator:NETWork:COUNt?

Description: Query only. The query outputs the number of embedding/de-embedding networks on the indicated channel.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: NA

Default Value: 0

Syntax Example: `:CALC1:FSIM:NETW:COUN?`

:CALCulate{1-16}:FSIMulator:NETWork:DIELectric <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric?

Description: The command sets the current T-Line network other dielectric value on the indicated channel. The query outputs the current T-Line network dielectric value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:DIEL 2.5`

`:CALC1:FSIM:NETW:DIEL?`

:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:EVEN <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:EVEN?

Description: Set the current network other dielectric even value on the indicated channel. The query outputs the current network dielectric even value on the indicated channel.

For the purposes of entering line information, the MS4640B VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term "common-mode" is also used for even mode. The term "differential-mode" is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR3> The output parameter is unitless number.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:DIEL:EVEN 1.2E0`

`:CALC1:FSIM:NETW:DIEL:EVEN?`

:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:ODD <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:ODD?

Description: Set the current network other dielectric odd value on the indicated channel. The query outputs the current network dielectric odd value on the indicated channel.

For the purposes of entering line information, the MS4640B VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term "common-mode" is also used for even mode. The term "differential-mode" is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR3> The output parameter is unitless number.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:DIEL:ODD 1.2E0`

`:CALC1:FSIM:NETW:DIEL:ODD?`

:CALCulate{1-16}:FSIMulator:NETWork:FREQuency <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:FREQuency?

Description: The command sets the current T-Line network line loss frequency value on the indicated channel. The query outputs the current T-Line network line loss frequency value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Hertz.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:FREQ 1E7`

`:CALC1:FSIM:NETW:FREQ?`

:CALCulate{1-16}:FSIMulator:NETWork:L <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:L?

Description: The command sets the current LC network inductance value on the indicated channel. The query outputs the current LC network inductance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:L 3.0E-9`

`:CALC1:FSIM:NETW:L?`

:CALCulate{1-16}:FSIMulator:NETWork:LENGth <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:LENGth?

Description: The command sets the current T-Line network line length value on the indicated channel. The query outputs the current T-Line network line length value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:LENG 2.5E-2`

`:CALC1:FSIM:NETW:LENG?`

:CALCulate{1-16}:FSIMulator:NETWork:LOSS <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:LOSS?

Description: The command sets the current T-Line network line loss value on the indicated channel. The query outputs the current T-Line network line loss value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: NA

Query Output: <NR3> The output parameter is in dB/mm.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:LOSS 3.0E0`

`:CALC1:FSIM:NETW:LOSS?`

:CALCulate{1-16}:FSIMulator:NETWork:LOSS:EVEN <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:LOSS:EVEN?

Description: Set the current network line loss even value on the indicated channel. The query outputs the current network line loss even value on the indicated channel.

For the purposes of entering line information, the MS4640B VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: NA

Query Output: <NR3> The output parameter is in dB/mm.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:LOSS:EVEN 3.0E0`

`:CALC1:FSIM:NETW:LOSS:EVEN?`

:CALCulate{1-16}:FSIMulator:NETWork:LOSS:ODD <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:LOSS:ODD?

Description: Set the current network line loss odd value on the indicated channel. The query outputs the current network line loss odd value on the indicated channel.

For the purposes of entering line information, the MS4640B VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled-line pair can be driven:

- In phase (the even mode, also called common-mode) or
- 180 degrees out of phase (the odd mode, also called differential-mode) or
- Any combination of those modes.

The term "common-mode" is also used for even mode. The term "differential-mode" is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: NA

Query Output: <NR3> The output parameter is in dB/mm.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:LOSS:ODD 3.0E0`

`:CALC1:FSIM:NETW:LOSS:ODD?`

:CALCulate{1-16}:FSIMulator:NETWork:MODe <char>
:CALCulate{1-16}:FSIMulator:NETWork:MODe?

Description: The command sets the current network embed/de-embed mode on the indicated channel. The query outputs the current network embed/de-embed mode on the indicated channel.

Cmd Parameters: <char> EMBed | DEEMbed

Query Parameters: NA

Query Output: <char> EMB | DEEM

Range: NA

Default Value: EMB

Syntax Example: `:CALC1:FSIM:NETW:MOD EMB`

`:CALC1:FSIM:NETW:MOD?`

:CALCulate{1-16}:FSIMulator:NETWork:PORT <char>
:CALCulate{1-16}:FSIMulator:NETWork:PORT?

Description: The command sets the current network port number on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. The query outputs the current network port number on the indicated channel.

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4 | PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Query Parameters: NA

Query Output: <char> PORT1 | PORT2 | PORT3 | PORT4 | PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:FSIM:NETW:PORT PORT1`

`:CALC1:FSIM:NETW:PORT?`

:CALCulate{1-16}:FSIMulator:NETWork:R <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:R?

Description: The command sets the current R network resistance value on the indicated channel. The query outputs the current R network resistance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:R 7.5E1`

`:CALC1:FSIM:NETW:R?`

:CALCulate{1-16}:FSIMulator:NETWork:S2P <string>
:CALCulate{1-16}:FSIMulator:NETWork:S2P?

Description: The command sets the current network S2P filename on the indicated channel. The query outputs the current network S2P filename on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist. See Chapter 2, “Programming the VectorStar Series VNA”, “Notational Conventions” on page 2-7 for more information.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:FSIM:NETW:S2P 'C:\filename.s2p'`

`:CALC1:FSIM:NETW:S2P?`

:CALCulate{1-16}:FSIMulator:NETWork:S4P <string>
:CALCulate{1-16}:FSIMulator:NETWork:S4P?

Description: Set the current network S4P Filename on the indicated channel. The query outputs the current network S4P Filename on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s4p' where x:\directory\filename.s2p must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: NA

Query Output: <string>

Range: NA

Default Value: NA

Syntax Example:
```
:CALC1:FSIM:NETW:S4P 'C:\filename.s4p'
:CALC1:FSIM:NETW:S4P?
```

:CALCulate{1-16}:FSIMulator:NETWork:S4P:PORTs <char1>, <char2>, <char3>, <char4>
:CALCulate{1-16}:FSIMulator:NETWork:S4P:PORTs?

Description: This command requires a 4-Port VNA instrument. Sets the port assignments for the current S4P network to be embedded/de-embedded on the channel indicated. The query outputs the port assignments for the current S4P network to be embedded/de-embedded on the channel indicated.

The port assignments are set by four <char> values where:

- The <char1> value sets the Port 1 assignment.
- The <char2> value sets the Port 2 assignment.
- The <char3> value sets the Port 3 assignment.
- The <char4> value sets the Port 4 assignment.
- When considered as a set, the value of each <char> must be unique.

Cmd Parameters: <char1> PORT1 | PORT2 | PORT3 | PORT4

<char2> PORT1 | PORT2 | PORT3 | PORT4

<char3> PORT1 | PORT2 | PORT3 | PORT4

<char4> PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: NA

Query Output: <char> PORT1 | PORT2 | PORT3 | PORT4

Syntax Example:
```
:CALC1:FSIM:NETW:S4P:PORT PORT1, PORT4, PORT3, PORT2
:CALC1:FSIM:NETW:S4P:PORT?
```

:CALCulate{1-16}:FSIMulator:NETWork:S4P:TERM:IGNore <char1>, {<char2>, ..., <charn>}
:CALCulate{1-16}:FSIMulator:NETWork:S4P:TERM:IGNore?

Description: This command requires a 4-Port VNA instrument. The command sets one or more S-Parameter terms to ignore from the current S4P network to be embedded/de-embedded on the channel indicated. At least one S-Parameter must be specified. Up to 16 S-parameters can be specified. The query outputs the S-Parameter terms to ignore from the current S4P network to be embedded/de-embedded on the channel indicated.

Cmd Parameters: <char> S11 | S12 | S21 | S22 | S13 | S31 | S23 | S32 | S33 | S14 | S24 | S34 | S41 | S42 | S43 | S44

Query Parameters: NA

Query Output: <char> S11 | S12 | S21 | S22 | S13 | S31 | S23 | S32 | S33 | S14 | S24 | S34 | S41 | S42 | S43 | S44

Range: NA

Default: NA

Syntax Example: `:CALC1:FSIM:NETW:S4P:TERM:IGN S11, S22, S33, S44`

`:CALC1:FSIM:NETW:S4P:TERM:IGN?`

:CALCulate{1-16}:FSIMulator:NETWork:S4P:TRANsmission:TERM <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:S4P:TRANsmission:TERM?

Description: This command requires a 4-Port VNA instrument. Set the current network S4P transmission terms value to 1 (one) or 0 (zero) on the indicated channel. The query outputs the current network S4P transmission term set value on the indicated channel.

Cmd Parameters: <NRf> Input parameter is unitless number either 1 or 0.

Query Parameters: NA

Query Output: <NR1> Output parameter is a unitless number.

Range: 0 or 1

Default Value: 0

Syntax Example: `:CALC1:FSIM:NETW:S4P:TRAN:TERM`

`:CALC1:FSIM:NETW:S4P:TRAN:TERM?`

:CALCulate{1-16}:FSIMulator:NETWork:SWAPs2p <char>
:CALCulate{1-16}:FSIMulator:NETWork:SWAPs2p?

Description: The command sets the current network swap S2P file data flag on the indicated channel. The query outputs the current network swap S2P file data flag on the indicated channel.

Cmd Parameters: <char> TRUE | FALSe | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: FALS

Syntax Example: `:CALC1:FSIM:NETW:SWAP TRUE`

`:CALC1:FSIM:NETW:SWAP?`

:CALCulate{1-16}:FSIMulator:NETWork:TYPe <char>
:CALCulate{1-16}:FSIMulator:NETWork:TYPe?

Description: The command sets the current network type on the indicated channel. The query outputs the current network type on the indicated channel. The available network choices depend on whether the instrument is in 2-Port or 4-Port VNA mode. All 2-Port networks are available for 4-Port VNAs. The following network types are available:

Types Available for 2-Port VNA Instruments

If the instrument is in two-port mode, the following types are available:

- LS = 2-Port or 4-Port VNAs. Series inductance
- LP = 2-Port or 4-Port VNAs. Parallel inductance
- CS = 2-Port or 4-Port VNAs. Series capacitance
- CP = 2-Port or 4-Port VNAs. Parallel capacitance
- RS = 2-Port or 4-Port VNAs. Resistive series network.
- RP = 2-Port or 4-Port VNAs. Resistive parallel network.
- TLine = 2-Port or 4-Port VNAs. A defined transmission line with specifications for Impedance (Ohms), Length (Meters), Loss (dB/mm), @ Frequency (GHz), and Dielectric Value. Note that programmatically, length is entered in Meters. From the user interface, length is usually entered in millimeters.
- S2Pfile = 2-Port or 4-Port VNAs. Allows an S2P calibration file to be used.

Types Available for 4-Port VNA Instruments

If the instrument is in four-port mode, all of the network types above are available with the addition of the following network types:

- S4Pfile = 4-Port VNAs only. Allows an S4P calibration file to be used.
- LCKTFour = 4-Port VNAs only. A four-node inductance L circuit. Port assignments are defined in separate commands.
- CCKTFour = 4-Port VNAs only. A four-node capacitance C circuit. Port assignments are defined in separate commands.
- TLINEFour = 4-Port VNAs only. Allows two separate through (“thru”) lines to be used. In separate commands, each line is defined by Length (Meters), @ Frequency (GHz), Z0-Odd (Ohms), Loss-Odd (dB/mm), Dielectric Odd (unitless number), Z0Even (Ohms), Loss-Even (dB/mm), and Dielectric Even (unitless number). Note that programmatically, length is entered in Meters. From the user interface, length is usually entered in millimeters.
- RCKTFour = 4-Port VNAs only. A four-node resistive R circuit. Port assignments are defined in separate commands.

Cmd Parameters: <char> LS | LP | CS | CP | RS | RP | TLine | S2Pfile | S4Pfile | LCKTFour | CCKTFour | TLINEFour | RCKTFour

Query Parameters: NA

Query Output: <char> LS | LP | CS | CP | RS | RP | TL | S2P | S4P | LCKTF | CCKTF | TLINEF | RCKTF

Range: NA

Default Value: S4Pfile

Syntax Example:
```
:CALC1:FSIM:NETW:TYP S4PFILE
:CALC1:FSIM:NETW:TYP?
```

:CALCulate{1-16}:FSIMulator:NETWork:Z0 <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:Z0?

Description: The command sets the current T-Line network impedance Z0 (Z zero) value on the indicated channel. The query outputs the current T-Line network impedance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: 50.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:Z0 7.5E1`

`:CALC1:FSIM:NETW:Z0?`

:CALCulate{1-16}:FSIMulator:NETWork:Z0:EVEN <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:Z0:EVEN?

Description: Set the current network impedance Z0 (Z zero) even value on the indicated channel. The query outputs the current network impedance even value on the indicated channel. For the purposes of entering line information, the MS463xA/MS464xA Series VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Ohms.

Range: MPND

Default: 50.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:Z0:EVEN 7.5E1`

`:CALC1:FSIM:NETW:Z0:EVEN?`

:CALCulate{1-16}:FSIMulator:NETWork:Z0:ODD <NRf>
:CALCulate{1-16}:FSIMulator:NETWork:Z0:ODD?

Description: Set the current network impedance odd value on the indicated channel. The query outputs the current network impedance odd value on the indicated channel. For the purposes of entering line information, the MS463xA/MS464xA Series VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Ohms.

Range: MPND

Default: 50.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW:Z0:ODD 7.5E1`

`:CALC1:FSIM:NETW:Z0:ODD?`

:CALCulate{1-16}:FSIMulator:NETWork[:STATe] <char>
:CALCulate{1-16}:FSIMulator:NETWork[:STATe]?

Description: The command sets the network embedding/de-embedding function on/off state on the indicated channel. The query outputs the network embedding/de-embedding function on/off state on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:FSIM:NETW ON`

`:CALC1:FSIM:NETW?`

5-14 :CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation

The :CALCulate{1-16}:FSIMulator:NETWork{1-50} subsystem uses existing calibration files with a simulated network of various types to evaluate predicted performance. The commands use index numbers to identify the appropriate network.

Calibration Simulation Subsystems

These subsystems are used to create a calibrated state in the instrument which is followed by adding the required error correction coefficients for the required calibration type. If this approach is used, each error correction coefficient is entered by separate commands. Simulated calibration subsystems are:

- ":CALCulate{1-16}:FSIMulator:NETWork Subsystem - Simulation" on page 5-51
- ":CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation" on page 5-63
- ":CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation" on page 5-63
- ":SENSe{1-16}:CORRection:COEFficient:PORT Subsystem - Simulation" on page 5-269
- ":SENSe{1-16}:CORRection:COEFficient Subsystem - Simulation" on page 5-274

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:C <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:C?

Description: The command modifies the indicated LC network capacitance value on the indicated channel. The query outputs the indicated LC network capacitance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: <NR3> The output parameter is in Farads.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:C 3.0E-12`

`:CALC1:FSIM:NETW1:C?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DELete

Description: The command deletes the indicated network from the indicated channel. No query.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:FSIM:NETW1:DEL`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric?

Description: The command modifies the indicated T-Line network other dielectric value on the indicated channel. The query outputs the indicated T-Line network other dielectric value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:DIEL 2.5E0`

`:CALC1:FSIM:NETW1:DIEL?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:EVEN <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:EVEN?

Description: Modify the indicated network other dielectric even value on the indicated channel. The query outputs the indicated network dielectric even value on the indicated channel.

For the purposes of entering line information, the MS463xA/MS464xA Series VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: See Chapter 2, “Programming the VectorStar Series VNA”, “Calibration Component Parameters” on page 2-38.

Syntax Example: `:CALC1:FSIM:NETW1:DIEL:EVEN 1.2E0`

`:CALC1:FSIM:NETW1:DIEL:EVEN?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:ODD <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:ODD?

Description: Modify the indicated network other dielectric odd value on the indicated channel. The query outputs the indicated network dielectric odd value on the indicated channel.

For the purposes of entering line information, the MS4640B VNA uses an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:CALC1:FSIM:NETW1:DIEL:ODD 1.2E0`

`:CALC1:FSIM:NETW1:DIEL:ODD?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:FREQuency <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:FREQuency?

Description: The command modifies the indicated T-Line network line loss frequency value on the indicated channel. The query outputs the indicated T-Line network line loss frequency value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:FREQ 1.0E4`

`:CALC1:FSIM:NETW1:FREQ?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:L <NRf>
:CALCulate{1-16}:FSIMulator:N

ETWork{1-50}:L?

Description: The command modifies the indicated LC network inductance value on the indicated channel. The query outputs the indicated LC network inductance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:L 5.0E-9`

`:CALC1:FSIM:NETW1:L?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LENGth <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LENGth?

Description: The command modifies the indicated T-Line network line length value on the indicated channel. The query outputs the indicated T-Line network line length value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:LENG 2.5E-2`

`:CALC1:FSIM:NETW1:LENG?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS?

Description: The command modifies the indicated T-Line network line loss value on the indicated channel. The query outputs the indicated T-Line network line loss value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: <NR3> The output parameter is in dB/mm.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:LOSS 3.0E0`

`:CALC1:FSIM:NETW1:LOSS?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:EVEN <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:EVEN?

Description: Modify the indicated network line loss even value on the indicated channel. The query output the indicated network line loss even value on the indicated channel.

For the purposes of entering line information, the MS4640B VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: <NR3> The output parameter is in dB/mm.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:LOSS:EVEN 3.0E0`

`:CALC1:FSIM:NETW1:LOSS:EVEN?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:ODD <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:ODD?

Description: Modify the indicated network line loss odd value on the indicated channel. The query outputs the indicated network line loss odd value on the indicated channel.

For the purposes of entering line information, the MS464xB Series VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: <NR3> The output parameter is in dB/mm.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:LOSS:ODD 3.0E0`

`:CALC1:FSIM:NETW1:LOSS:ODD?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:MODe <char>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:MODe?

Description: The command modifies the indicated network embed/de-embed mode on the indicated channel. The query outputs the indicated network embed/de-embed mode on the indicated channel.

Cmd Parameters: <char> EMBed | DEEMbed

Query Parameters: <char> EMB | DEEM

Range: NA

Default Value: EMB

Syntax Example: `:CALC1:FSIM:NETW1:MOD EMB`

`:CALC1:FSIM:NETW1:MOD?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:PORT <char>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:PORT?

Description: The command modifies the indicated network port number on the indicated channel. The query outputs the indicated network port number on the indicated channel.

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4 | PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Query Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4 | PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:FSIM:NETW1:PORT PORT1`

`:CALC1:FSIM:NETW1:PORT?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:R <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:R?

Description: The command modifies the indicated R network resistance value on the indicated channel. The query outputs the indicated R network resistance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:R 7.5E1`

`:CALC1:FSIM:NETW1:R?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S2P <string>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S2P?

Description: The command modifies the indicated network S2P filename on the indicated channel. The query outputs the indicated network S2P filename on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\filename.s2p must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: <char> Filename and path in the form: x:\directory\filename.s2p

Range: NA

Default Value: NA

Syntax Example: `:CALC1:FSIM:NETW1:S2P 'C:\filename.s2p'`

`:CALC1:FSIM:NETW1:S2P?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P <string>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P?

Description: Modify the indicated network S4P filename on the indicated channel. The query outputs the indicated network S4P filename on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s4p' where x:\directory\filename.s2p must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: <string>

Range: NA

Default Value: NA

Syntax Example: `:CALC1:FSIM:NETW1:S4P 'C:\filename.s4p'`

`:CALC1:FSIM:NETW1:S4P?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:PORTs <char>, <char>, <char>, <char>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:PORTs?

Description: Set the port assignments for the indicated S4P network to be embedded/de-embedded on the indicated channel. The command requires a 4-Port VNA instrument. The query outputs the port assignments for the indicated S4P network to be embed/de-embed on the channel indicated.

The first entered port number is for Port 1, the second for Port 2, the third for Port 3, and the fourth for Port 4. For the Syntax Example below, to assign Port 2, Port 3, Port 1, and Port 4, the command is:

```
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:PORTs PORT2, PORT3,
PORT1, PORT4
```

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default: NA

Syntax Example: `:CALC1:FSIM:NETW1:S4P:PORT PORT2,PORT3,PORT1,PORT4`

`:CALC1:FSIM:NETW1:S4P:PORT?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TERM:IGNore <char> {,<char2>, ...,<charn>}
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TERM:IGNore?

Description: Set the S-Parameters to ignore from the indicated S4P network to be embed/de-embed on the channel indicated. At least one S-Parameter to ignore must be defined. Up to 16 S-Parameters to ignore can be defined. a 4-Port VNA instrument is required. The query outputs the S-Parameters to ignore from the indicated S4P network to be embedded/de-embeded on the channel indicated.

Cmd Parameters: <char> S11 | S12 | S21 | S22 | S13 | S31 | S23 | S32 | S33 | S14 | S24 | S34 | S41 | S42 | S43 | S44

Query Parameters: <char> S11 | S12 | S21 | S22 | S13 | S31 | S23 | S32 | S33 | S14 | S24 | S34 | S41 | S42 | S43 | S44

Range: NA

Default: NA

Syntax Example: `:CALC1:FSIM:NETW1:S4P:TERM:IGN S11, S22, S33, S44`

`:CALC1:FSIM:NETW1:S4P:TERM:IGN?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TRANsmission:TERM <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TRANsmission:TERM?

Description: Set the current network S4P Transmission Terms value to 1 (one) or 0 (zero) on the indicated channel. The query outputs the current network S4P transmission term set value on the indicated channel.

Cmd Parameters: <NRf> Input parameter is unitless number either 1 or 0.

Query Parameters: <NR1> Output parameter is a unitless number.

Range: 0 or 1

Default Value: 0

Syntax Example: `:CALC1:FSIM:NETW1:S4P:TRAN:TERM`

`:CALC1:FSIM:NETW1:S4P:TRAN:TERM?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:SWAPs2p <char>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:SWAPs2p?

Description: The command modifies the indicated network swap S2P file data flag on the indicated channel. The query outputs the indicated network swap S2P file data flag on the indicated channel.

Cmd Parameters: <char> TRUE | FALSe | 1 | 0

Query Parameters: <char> 1 | 0

Range: NA

Default Value: FALS

Syntax Example: `:CALC1:FSIM:NETW1:SWAP TRUE`

`:CALC1:FSIM:NETW1:SWAP?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:TYPe <char>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:TYPe?

Description: On the indicated channel, the command modifies the indicated network type. The query outputs the indicated network type on the indicated channel. The available network choices depend on whether the instrument is in 2-Port or 4-Port VNA mode. All 2-Port networks are available for 4-Port VNAs. The following network types are available:

Types Available for 2-Port VNA Instruments

If the instrument is in two-port mode, the following types are available:

- LS = 2-Port or 4-Port VNAs. Series inductance
- LP = 2-Port or 4-Port VNAs. Parallel inductance
- CS = 2-Port or 4-Port VNAs. Series capacitance
- CP = 2-Port or 4-Port VNAs. Parallel capacitance
- RS = 2-Port or 4-Port VNAs. Resistive series network.
- RP = 2-Port or 4-Port VNAs. Resistive parallel network.
- TLine = 2-Port or 4-Port VNAs. A defined transmission line with specifications for Impedance (Ohms), Length (Meters), Loss (dB/mm), @ Frequency (GHz), and Dielectric Value. Note that programmatically, length is entered in Meters. From the user interface, length is usually entered in millimeters.
- S2Pfile = 2-Port or 4-Port VNAs. Allows an S2P calibration file to be used.

Types Available for 4-Port VNA Instruments

If the instrument is in four-port mode, all of the network types above are available with the addition of the following network types:

- S4Pfile = 4-Port VNAs only. Allows an S4P calibration file to be used.
- LCKTFour = 4-Port VNAs only. A four-node inductance L circuit. Port assignments are defined in separate commands.
- CCKTFour = 4-Port VNAs only. A four-node capacitance C circuit. Port assignments are defined in separate commands.
- TLINEFour = 4-Port VNAs only. Allows two separate through (“thru”) lines to be used. In separate commands, each line is defined by Length (Meters), @ Frequency (GHz), Z0-Odd (Ohms), Loss-Odd (dB/mm), Dielectric Odd (unitless number), Z0Even (Ohms), Loss-Even (dB/mm), and Dielectric Even (unitless number). Note that programmatically, length is entered in Meters. From the user interface, length is usually entered in millimeters.
- RCKTFour = 4-Port VNAs only. A four-node resistive R circuit. Port assignments are defined in separate commands.

Cmd Parameters: <char> LS | LP | CS | CP | RS | RP | TLine | S2Pfile | S4Pfile | LCKTFour | CCKTFour | TLINEFour | RCKTFour

Query Parameters: <char> LS | LP | CS | CP | RS | RP | TL | S2P | S4P | LCKTF | CCKTF |TLINEF | RCKTF

Range: NA

Default Value: LSCP

Syntax Example: `:CALC1:FSIM:NETW1:TYP TLine`

`:CALC1:FSIM:NETW1:TYP?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0 <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0?

Description: The command modifies the indicated T-Line network impedance value on the indicated channel. The query outputs the indicated T-Line network impedance value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in ohms.

Query Parameters: <NR3> The output parameter is in ohms.

Range: MPND

Default Value: 50.00000000000E+000

Syntax Example: `:CALC1:FSIM:NETW1:Z0 7.5E1`

`:CALC1:FSIM:NETW1:Z0?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:EVEN <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:EVEN?

Description: Modify the indicated network impedance even value on the indicated channel. The query outputs the indicated network impedance even value on the indicated channel. For the purposes of entering line information, the MS4640B VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in ohms.

Query Parameters: <NR3> The output parameter is in ohms.

Range: MPND

Default: 5.00000000000E+001

Syntax Example: `:CALC1:FSIM:NETW1:Z0:EVEN 7.5E1`

`:CALC1:FSIM:NETW1:Z0:EVEN?`

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:ODD <NRf>
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:ODD?

Description: Modify the indicated network impedance odd value on the indicated channel. The query outputs the indicated network impedance odd value on the indicated channel. For the purposes of entering line information, the MS4640B VNAs use an even/odd mode formalism as is consistent with many circuit simulators. The central concept is that a coupled line pair can be driven in phase (the even mode) or 180 degrees out of phase (the odd mode) or any combination of those modes. The term “common-mode” is also used for even mode. The term “differential-mode” is also used for odd mode. In the case of very weak coupling where C_x is close to 0, these modes see the same impedances, same losses, and same phase velocities so there is no need to use this mode separation. As the coupling increases, at the very least, the impedances seen by these two modes diverge requiring two impedance entries where the effective capacitances seen by the conductors in the two modes are clearly different. That is the end of changes for symmetric TEM systems, where this approach will work for common coax, stripline and some microstrip cases.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default: 5.00000000000E+001

Syntax Example:

```
:CALC1:FSIM:NETW1:Z0:ODD 7.5E1
:CALC1:FSIM:NETW1:Z0:ODD?
```

5-15 :CALCulate{1-16}:IMPedance:TRANsformation Subsystem

The :CALCulate{1-16}:IMPedance:TRANsformation subsystem commands set configuration parameters for impedance transformation.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:R0 <NRf>
:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:R0?

Description: The command sets the resistive value of the impedance transformation on the given port and channel. If the channel is not specified, applies the resistive impedance transformation to the active port on the active channel. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. The query outputs the resistive value of the impedance transformation on the given port and channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: 0

Syntax Example: `:CALC1:IMP:TRAN:PORT1:R0 25.5`

`:CALC1:IMP:TRAN:PORT1:R0?`

:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:X0 <NRf>
:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:X0?

Description: Sets the reactive value of the impedance transformation on the given port and channel. The query outputs the reactive value of the impedance transformation on the given port and channel.

Cmd Parameters: <NRf> The input parameter is in Ohms

Query Parameters: <NR3> The input parameter is in Ohms.

Range: MPND

Default Value: NA

Syntax Example: `:CALC:IMP:TRAN:PORT:X0 1E3`

`:CALC:IMP:TRAN:PORT:X0?`

:CALCulate{1-16}:IMPedance:TRANsformation[:STATe]
:CALCulate{1-16}:IMPedance:TRANsformation[:STATe]?

Description: The command toggles the impedance transformation on and off on the indicated channel. The query returns the impedance transformation on/off status for the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: :CALC:IMP:TRAN ON

:CALC:IMP:TRAN?

5-16 :CALCulate{1-16}:MARKer Subsystem

The :CALCulate{1-16}:MARKer subsystem commands provide control of the marker table display and marker coupling.

Marker Subsystems

Related marker configuration, control, and reporting commands are described in the following subsystem sections:

- ":CALCulate{1-16}:DISPlay:MARKer Subsystem" on page 5-17
- ":CALCulate{1-16}:MARKer Subsystem" on page 5-76
- ":CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem" on page 5-107
- ":CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem" on page 5-110
- ":CALCulate{1-16}[:SELected]:MARKer Subsystem" on page 5-165
- ":CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem" on page 5-178
- ":DISPlay Subsystem" on page 5-223

:CALCulate:MARKer:TABle[:STATe] <char>
:CALCulate:MARKer:TABle[:STATe]?

Description: The command turns the marker table display on/off. The query outputs the marker table display on/off status.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC:MARK:TAB ON`

`:CALC:MARK:TAB?`

:CALCulate{1-16}:MARKer:COUPle <char>
:CALCulate{1-16}:MARKer:COUPle?

Description: The command toggles marker coupling on/off on the given channel. The query outputs the marker coupling on/off state on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:COUP ON`

`:CALC1:MARK:COUP?`

5-17 :CALCulate{1-16}:MXP Subsystem

The :CALCulate{1-16} MXP subsystem commands control the mixed-mode measurement technique.

Measurement and Mixed Mode Subsystems

- ":CALCulate{1-16}:MXP Subsystem" on page 5-77
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101

General Parameters

The general command parameters are:

- :D1S0 sets the port mapping to one differential pair and no singleton.
- :D1S1 sets the port mapping to one differential pair and one singleton.
- :D1S2 sets the port mapping to one differential pair and two singletons.
- :D2S0 sets the port mapping to two differential ports and no singletons.

:CALCulate{1-16}:MXP:D1S0:TOPology <char>
:CALCulate{1-16}:MXP:D1S0:TOPology?

Description: For 4-Port VNA instrument only. The command sets the D1S0 (D One S Zero) Device actual port mapping for one differential pair and no singleton when forming an MXP data file for the indicated channel. The <char> parameter establishes the mappings between the DUT differential-pair ports and the VNA ports:

- MAP12 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 2.
- MAP21 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 1.
- MAP13 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 3.
- MAP31 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 1.
- MAP23 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 3.
- MAP32 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 2.
- MAP14 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 4.
- MAP41 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 1.
- MAP24 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 4.
- MAP42 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 2.
- MAP34 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 4.
- MAP43 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 3.
- NONE = Returned in the query only if no DUT Port to VNA Port mapping has been established.

The query outputs the D1S0 Device actual ports mapping when forming an MXP data file for the indicated channel and trace.

Cmd Parameters: <char> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

Query Parameters: <char> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

Range: NA

Default: MAP12

Syntax Example:
```
:CALC1:MXP:D1S0:TOP MAP12
:CALC1:MXP:D1S0:TOP?
```

:CALCulate{1-16}:MXP:D1S1:TOPology <char1>, <char2>
:CALCulate{1-16}:MXP:D1S1:TOPology?

Description: For 4-Port VNA instruments only. The command sets the D1S1 (D one S one) Device actual ports mapping for one differential pair and one singleton when forming an MXP data file for the indicated channel. Note that port mappings must be port exclusive.

<char1> Parameter

The <char1> parameter establishes the mappings between the DUT differential-pair ports and the VNA ports:

- MAP12 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 2.
- MAP21 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 1.
- MAP13 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 3.
- MAP31 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 1.
- MAP23 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 3.
- MAP32 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 2.
- MAP14 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 4.
- MAP41 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 1.
- MAP24 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 4.
- MAP42 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 2.
- MAP34 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 4.
- MAP43 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 3.
- NONE = Returned in the query only if no DUT Port to VNA Port mapping has been established.

<char2> Parameter

The <char2> parameter establishes the mapping between the DUT singleton port and the VNA port:

- MAP1 = DUT Singleton to VNA Port 1.
- MAP2 = DUT Singleton to VNA Port 2.
- MAP3 = DUT Singleton to VNA Port 3.
- MAP4 = DUT Singleton to VNA Port 4.
- NONE = Returned in the query only if no DUT singleton mapping has been established.

The query outputs the D1S1 Device actual ports mapping when forming an MXP data file for the indicated channel and trace.

Cmd Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

<char2> MAP1 | MAP2 | MAP3 | MAP4

Query Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

<char2> MAP1 | MAP2 | MAP3 | MAP4 | NONE

Range: NA

Default: MAP12,MAP3

Syntax Example: `:CALC1:MXP:D1S1:TOP MAP32,MAP1`

`:CALCulate{1-16}:MXP:D1S1:TOP?`

:CALCulate{1-16}:MXP:D1S2:TOPology <char1>, <char2>, <char3>
:CALCulate{1-16}:MXP:D1S2:TOPology?

Description: For 4-Port VNA instruments only. The command sets the D1S2 (D one S two) Device actual ports mapping to one differential pair and two singletons when forming an MXP data file for the indicated channel. Note that port mappings must be port exclusive.

<char1> Parameter

The <char1> parameter establishes the mappings between the DUT differential-pair ports and the VNA ports:

- MAP12 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 2.
- MAP21 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 1.
- MAP13 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 3.
- MAP31 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 1.
- MAP23 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 3.
- MAP32 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 2.
- MAP14 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 4.
- MAP41 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 1.
- MAP24 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 4.
- MAP42 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 2.
- MAP34 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 4.
- MAP43 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 3.
- NONE = Returned in the query only if no DUT Port to VNA Port mapping has been established.

<char2> Parameter

The <char2> parameter establishes the mapping between the first DUT singleton and the VNA port:

- MAP1 = First DUT Singleton to VNA Port 1.
- MAP2 = First DUT Singleton to VNA Port 2.
- MAP3 = First DUT Singleton to VNA Port 3.
- MAP4 = First DUT Singleton to VNA Port 4.
- NONE = Returned in the query only if the first DUT singleton has not been mapped.

<char3> Parameter

The <char3> parameter establishes the mapping between the second DUT singleton and the VNA port:

- MAP1 = Second DUT Singleton to VNA Port 1.
- MAP2 = Second DUT Singleton to VNA Port 2.
- MAP3 = Second DUT Singleton to VNA Port 3.
- MAP4 = Second DUT Singleton to VNA Port 4.
- NONE = Returned in the query only if the second DUT singleton has not been mapped.

The query outputs the D1S2 Device actual ports mapping when forming an MXP data file for the indicated channel and trace.

Cmd Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

<char2> MAP1 | MAP2 | MAP3 | MAP4

<char3> MAP1 | MAP2 | MAP3 | MAP4

Query Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

<char2> MAP1 | MAP2 | MAP3 | MAP4 | NONE

<char3> MAP1 | MAP2 | MAP3 | MAP4 | NONE

Range: NA

Default: MAP12,MAP2,MAP4

Syntax Example: `:CALC1:MXP:D1S2:TOP MAP32,MAP4,MAP1`

`:CALC1:MXP:D1S2:TOP?`

:CALCulate{1-16}:MXP:D2S0:TOPology <char1>, <char2>
:CALCulate{1-16}:MXP:D2S0:TOPology?

Description: For 4-Port VNA instruments only. The command sets the D2S0 (D 2 S Zero) Device actual ports mapping as two differential ports and no singletons when forming an MXP data file for the indicated channel. Note that port mappings must be port exclusive.

<char1> Parameter

The <char1> parameter establishes the mappings between the first DUT differential-pair ports and the VNA ports:

- MAP12 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 2.
- MAP21 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 1.
- MAP13 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 3.
- MAP31 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 1.
- MAP23 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 3.
- MAP32 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 2.
- MAP14 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 4.
- MAP41 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 1.
- MAP24 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 4.
- MAP42 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 2.
- MAP34 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 4.
- MAP43 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 3.
- NONE = Returned in the query only if the first DUT Port Pair to the VNA Port Pair has not been mapped.

<char2> Parameter

The <char2> parameter establishes the mappings between the second DUT differential-pair ports and the VNA ports:

- MAP12 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 2.
- MAP21 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 1.
- MAP13 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 3.
- MAP31 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 1.
- MAP23 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 3.
- MAP32 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 2.
- MAP14 = DUT Port 1 to VNA Port 1, DUT Port 2 to VNA Port 4.
- MAP41 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 1.
- MAP24 = DUT Port 1 to VNA Port 2, DUT Port 2 to VNA Port 4.
- MAP42 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 2.
- MAP34 = DUT Port 1 to VNA Port 3, DUT Port 2 to VNA Port 4.
- MAP43 = DUT Port 1 to VNA Port 4, DUT Port 2 to VNA Port 3.
- NONE = Returned in the query only if the second DUT Port Pair to the VNA Port Pair has not been mapped.

Cmd Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

<char2> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

Query Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

<char2> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

Range: NA

Default: MAP12,MAP34

Syntax Example: `:CALC1:MXP:D2S0:TOP MAP24,MAP13`

`:CALC1:MXP:D2S0:TOP?`

5-18 :CALCulate{1-16}:NFIGure Subsystem

Use the Noise Figure commands to configure and measure noise figure values.

:CALCulate{1-16}:NFIGure:LOAD:DUT:S2P:DATA <char>

Description: For the indicated channel, the command loads the DUT S2P data from the identified file. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.S2P' where x:\directory\filename.S2P must exist. See definition of "<string>" on page 2-12.

Query Parameters: NA

Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:NFIG:LOAD:DUT:S2P:DATA 'c:\AnritsuVNA\duts2pdatafilename.s2p'`

:CALCulate{1-16}:NFIGure:NOISe:CALibration:PERForm?

Description: Query only. For the indicated channel, the command starts the noise figure calibration and then outputs the status on the given channel. Note that this command is only valid when the VNA is equipped with Option 41 – Noise Figure, and is in noise figure measurement mode. If the command is issued when the instrument is in Transmission/Reflection measurement mode, the command fails and returns a "1".

Cmd Parameters: NA

Query Parameters: <NR1> 0 | 1

Output: <NR1>, where:

- 0 = Successful completion of the calibration
- 1 = Error in the calibration

Range: NA

Default: NA

Syntax Example: `:CALC1:NFIG:NOIS:CAL:PERF?`

:CALCulate{1-16}:NFIGure:RMS:POINts <NRf>
:CALCulate{1-16}:NFIGure:RMS:POINts?

Description: For the indicated channel, the command sets the number of RMS (Root Mean Square) measurement points to be used in the noise power computation which sets the number of RMS measurement points for each sweep point. The query returns the number of measurements that were used in the noise power computation. Note that the noise power measurements are for each sweep point. For example, if 3,000 RMS points and 51 sweep points are set, there are then 3,000 measurements for each of the 51 sweep points for a total of 3,000 × 51 = 153,000 total measurements.

Cmd Parameters: <NRf>

Query Parameters: <NR3>

Output: <NR3>

Range: MPND

Default: NA

Syntax Example: `:CALC1:NFIG:RMS:POINts 3.000000E+003`

`:CALC1:NFIG:RMS:POIN?`

:CALCulate{1-16}:NFIGure:TEMPerature?

Description: For the indicated channel, the query returns the standard temperature of 290 K that was used for the cold termination measurements. The IEEE standard temperature definition is 290 K.

Cmd Parameters: NA

Query Parameters: <NR3>

Output: <NR3>

Range: MPND

Default: `2.900000E+002`

Syntax Example: `:CALC1:NFIG:TEMP?`

:CALCulate{1-16}:NFIGure:LOAD:RECVer:OFFSet:DATA <filespec>

Description: For the indicated channel, the command loads the Noise Figure Receiver cal offset data from the identified file.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.rco' where x:\directory\ must exist.

Query Parameters: NA

Query Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:NFIG:LOAD:RECV:OFFS:DATA c:\AnritsuVNA\rcal.rco`

:CALCulate{1-16}:NFIGure:NETWork:DEEMbed:FILe <string>
:CALCulate{1-16}:NFIGure:NETWork:DEEMbed:FILe?

Description: Sets the de-embedding network path for the .s2p file used for noise figure network extraction on the given channel. The query outputs the de-embedding network path for the .s2p file used for noise figure network extraction on the given channel.

Cmd Parameters: <string> "c:\nenffiles\myfile1.s2p"

Query Parameters: <char> c:\nenffiles\myfile1.s2p

Range: NA

Default: NA

Syntax Example: `:CALC1:NFIG:NETW:DEEM:FIL "c:\nenffiles\myfile1.s2p"`

`:CALC1:NFIG:NETW:DEEM:FIL?`

:CALCulate{1-16}:NFIGure:NETWork:DEEMbed:VALid?

Description: Outputs the state flag based on the Deembed Network Path for noise figure network extraction on the given channel.

Cmd Parameters: NA

Query Parameters: <NR1> Returns one of the following: 1 | 0

Returns 1 if the .s2p file is Valid.

Returns 0 if the .s2p file is Invalid

Range: NA

Default: NA

Syntax Example: `:CALC1:NFIG:NETW:DEEM:VAL?`

:CALCulate{1-16}:NFIGure:NETWork:DEEMbed[:STATe] <char>
:CALCulate{1-16}:NFIGure:NETWork:DEEMbed[:STATe]?

Description: Sets the Deembed state for noise figure network extraction on the given channel. The query outputs Deembed state for noise figure network extraction on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default: 0

Syntax Example: `:CALC1:NFIG:NETW:DEEM ON`

`:CALC1:NFIG:NETW:DEEM?`

:CALCulate{1-16}:NFIGure:NETWork:EMBed:FILe <string>
:CALCulate{1-16}:NFIGure:NETWork:EMBed:FILe?

Description: Sets the Embed Network Path for the for noise figure network extraction on the given channel. The query outputs the Embed Network Path for noise figure network extraction on the given channel.

Cmd Parameters: <string> "c:\nenffiles\myfile1.s2p"

Query Parameters: <char> c:\nenffiles\myfile1.s2p

Range: NA

Default: NA

Syntax Example: `:CALC1:NFIG:NETW:EMB:FIL "c:\nenffiles\myfile1.s2p"`

`:CALC1:NFIG:NETW:EMB:FIL?`

:CALCulate{1-16}:NFIGure:NETWork:EMBed:VALid?

Description: Outputs the state flag based on the Embed Network Path for noise figure network extraction on the given channel.

Cmd Parameters: NA

Query Parameters: <NR1> Returns one of the following: 1 | 0

Returns 1 if the .s2p file is Valid.

Returns 0 if the .s2p file is Invalid

Range: NA

Default: NA

Syntax Example: `:CALC1:NFIG:NETW:EMB:VAL?`

:CALCulate{1-16}:NFIGure:NETWork:EMBed[:STATe] <char>
:CALCulate{1-16}:NFIGure:NETWork:EMBed[:STATe]?

Description: Sets the Embed state for noise figure network extraction on the given channel. The query outputs the Embed state for noise figure network extraction on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default: 0

Syntax Example: `:CALC1:NFIG:NETW:EMB ON`

`:CALC1:NFIG:NETW:EMB?`

:CALCulate{1-16}:NFIGure:RECVer:OFFSet[:STATe] <char>
:CALCulate{1-16}:NFIGure:RECVer:OFFSet[:STATe]?

Description: Turns on/off the state of noise figure receiver cal offset of the indicated channel. The query outputs the on/off state of the noise figure receiver cal offset of the indicated channel

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Query Output: NA

Range: NA

Default: OFF

Syntax Example: `:CALC1:NFIG:RECV:OFFS:STAT ON`

`:CALC1:NFIG:RECV:OFFS:STAT?`

5-19 :CALCulate{1-16}:NXN Subsystem

The :CALCulate{1-16}:NXN subsystem commands provide combining three device pairings to determine the behavior of each individual device by combining the S2P file for each device.

Calibration Option Subsystems

Related calibration option configuration and control subsystems are:

- ":CALCulate{1-16}:CORRection Subsystem - Adapters/Merge Calibration" on page 5-14
- ":CALCulate{1-16}:EXTRaction Subsystem - Network Extraction" on page 5-33
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- ":SENSe{1-16}:CORRection:COLLect:HYBRid Subsystem" on page 5-290

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- ":CALCulate{1-16}:FORMat Subsystem - SnP Data" on page 5-49
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":FORMat Subsystem" on page 5-242
- ":HCOPy Subsystem" on page 5-245
- ":MMEMory Subsystem" on page 5-251

:CALCulate{1-16}:NXN:FILe <char>, <string>
:CALCulate{1-16}:NXN:FILe?

Description: The command sets the indicated NXN device or device-pair S2P filename on the indicated channel where the device refers to a single device or a device pair. An example is when defining a three-device system such as three mixers where one device becomes the calibration standard. The parameters are:

- DEV1 = Device 1
- DEV2 = Device 2
- DEV3 = Device 3
- DEV12 = Device 1 and Device 2
- DEV13 = Device 1 and Device 3
- DEV23 = Device 2 and Device 3

On the indicated channel, the query outputs the indicated NXN device or device pair S2P filename.

Cmd Parameters: <char> DEV1 | DEV2 | DEV3 | DEV12 | DEV13 | DEV23

<string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\filename.s2p must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: <char> DEV1 | DEV2 | DEV3 | DEV12 | DEV13 | DEV23

Range: NA

Default Value: NA

Syntax Example: `:CALC1:NXN:FIL DEV2,'c:\AnritsuVNA\nxnfilename.s2p'`

`:CALC1:NXN:FIL? DEV2`

:CALCulate{1-16}:NXN:IFPath:DEEMbedding[:STATe] <char>
:CALCulate{1-16}:NXN:IFPath:DEEMbedding[:STATe]?

Description: The command sets the NXN IF path de-embedding on/off state on the indicated channel. The query outputs the NXN IF path de-embedding on/off state on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:NXN:IFP:DEEM ON`

`:CALC1:NXN:IFP:DEEM?`

:CALCulate{1-16}:NXN:IFPath:DIRection <char>
:CALCulate{1-16}:NXN:IFPath:DIRection?

Description: The command sets the NXN IF path sweep direction on the indicated channel. The query outputs the NXN IF path sweep direction on the indicated channel.

Cmd Parameters: <char> FORward | REVerse

Query Parameters: <char> FOR | REV

Range: NA

Default Value: FOR

Syntax Example: `:CALC1:NXN:IFP:DIR REV`

`:CALC1:NXN:IFP:DIR?`

:CALCulate{1-16}:NXN:IFPath:FILe <string>
:CALCulate{1-16}:NXN:IFPath:FILe?

Description: The command sets the NXN IF path S2P filename on the indicated channel. The query outputs the NXN IF path S2P filename on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\filename.s2p must exist.

Query Parameters: <string> Filename and path in the form: x:\directory\filename.s2p

Range: NA

Default Value: NA

Syntax Example: `:CALC1:NXN:IFP:FIL 'C:\filename.s2p'`

`:CALC1:NXN:IFP:FIL?`

:CALCulate{1-16}:NXN:IFPath:MODified:FILe <string>
:CALCulate{1-16}:NXN:IFPath:MODified:FILe?

Description: The command sets the NXN Modified IF path S2P filename on the indicated channel. The query outputs the NXN Modified IF path S2P filename on the indicated channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist. See definition of "<string>" on page 2-12.

Query Parameters: <char> Filename and path in the form: x:\directory\filename.s2p

Range: NA

Default Value: NA

Syntax Example: `:CALC1:NXN:IFP:MOD:FIL 'C:\filename.s2p'`

`:CALC1:NXN:IFP:MOD:FIL?`

:CALCulate{1-16}:NXN:IFPath:MODified:SAVe

Description: The command saves the NXN Modified IF path S2P file on the indicated channel to where it was defined. The file name and path must already exist.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:NXN:IFP:MOD:FIL 'C:\filename.s2p'`

`:CALC1:NXN:IFP:MOD:SAV`

Related Cmd: `:CALCulate{1-16}:NXN:IFPath:FILe`

:CALCulate{1-16}:NXN:LENGth <char>, <NRf>
:CALCulate{1-16}:NXN:LENGth?

Description: The command sets the indicated NXN device length on the indicated channel where:

- DEV1 = Device 1
- DEV2 = Device 2
- DEV3 = Device 3

The query outputs the indicated NXN device length on the indicated channel.

Cmd Parameters: <char> DEV1 | DEV2 | DEV3

<NRf> The input parameter is in Meters.

Query Parameters: <char> DEV1 | DEV2 | DEV3

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:NXN:LENG DEV2, 2E-2`

`:CALC1:NXN:LENG? DEV2`

:CALCulate{1-16}:NXN:SOLVe <char>

Description: The command solves for the indicated NXN device and save the S2P file on the indicated channel where:

- DEV1 = Device 1
- DEV2 = Device 2
- DEV3 = Device 3

No query.

Cmd Parameters: <char> DEV1 | DEV2 | DEV3

Query Parameters: NA

Range: NA

Default Value: DEV1

Syntax Example: `:CALC1:NXN:SOLV DEV2`

5-20 :CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem

The :CALCulate{1-16}:PARameter subsystem commands control and report on the number of traces on the indicated channel. The :CALCulate{1-16}:PARameter{1-16} subsystem commands configure the types of parameters used and how they are displayed on each trace.

Trace Subsystems

Related trace and display subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem" on page 5-110
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}:PARameter:COUNt <NRf>
:CALCulate{1-16}:PARameter:COUNt?

Description: The command sets the number of traces on the specified channel.

The query outputs the number of traces on the specified channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 16

Default Value: 4

Syntax Example: `:CALC1:PAR:COUN 4`

`:CALC1:PAR:COUN?`

:CALCulate{1-16}:PARameter{1-16}:DEFine <char1> | <char1>,<char2> | <char1>,<char2>,<char3>,<char4>
:CALCulate{1-16}:PARameter{1-16}:DEFine?

Description: The command sets the measurement parameter of the indicated trace. Additional command parameters (<char2>, <char3>, and <char4>) may be required depending on the selection of the first <char> parameter. If multiple command parameters are required, each additional parameter is separated by a comma. Spaces between command parameters are ignored. The use of these parameters requires a 4-Port instrument: A3 | A4 | B3 | B4 | S13 | S23 | S33 | S31 | S32 | S14 | S24 | S34 | S41 | S42 | S43 | S44 | PORT3 | PORT4.

Available Measurement Parameters

The first parameter <char1> defines which S-Parameter, external analog input, or user-defined S-Parameter is measured. The available selections are:

- S11 = No other <char> parameters are required.
- S12 = No other <char> parameters are required.
- S13 = No other <char> parameters are required.
- S14 = No other <char> parameters are required.
- S21 = No other <char> parameters are required.
- S22 = No other <char> parameters are required.
- S23 = No other <char> parameters are required.
- S24 = No other <char> parameters are required.
- S31 = No other <char> parameters are required.
- S32 = No other <char> parameters are required.
- S33 = No other <char> parameters are required.
- S34 = No other <char> parameters are required.
- S41 = No other <char> parameters are required.
- S42 = No other <char> parameters are required.
- S43 = No other <char> parameters are required.
- S44 = No other <char> parameters are required.
- EXT1 = External <char2> is also required to define the Port number.
- EXT2 = <char2> is also required to define the Port number.
- USR = <char2> is also required to define the numerator,<char3> is also required to define the denominator, and <char4> is also required to define the Port number.
- MIXed = Response Mixed Mode, no other <char> parameters are required.
- NFIG = Noise Figure trace response, no other <char> parameters are required.
- NPOW = Noise Power trace response, no other <char> parameters are required.
- NTEMP = Noise Temperature trace response, no other <char> parameters are required.

Command Parameter Requirements when S11, S12, S21, or S22 is Selected

If the S11, S12, S21, S22, S13, S23, S33, S31, S32, S14, S24, S34, S41, S42, S43, S44, MIXed, NFIG, NPOW, or NTEMP parameters are selected for <char>:

- No additional command parameters are required.
- The use of Port 3 or Port 4 require a 4-Port instrument.
- For example, :CALC1:PAR1:DEF S11 sets the input-reflection coefficient measurement.
- The command syntax for S-Parameters:

  `:CALCulate{1-16}:PARameter{1-16}:DEFine <char1>`

Command Parameters Requirements when EXT1 or EXT2 is Selected

If an external analog input of EXT1 or EXT2 is selected for <char1>:

- An additional parameter of <char2> is required to define the power source port as PORT1, PORT2, PORT3, PORT4, L1, or L2.
- L1 is the same as Port 1 and L2 is the same as Port 2; both use Lightning 37xxxx syntax.
- The use of Port 3 or Port 4 require a 4-Port instrument.
- The command syntax for external analog input:

  `:CALCulate{1-16}:PARameter{1-16}:DEFine <char1>,<char2>`

Command Parameter Requirements when USR is Selected

If the user-defined S-Parameter of USR is selected for <char1>:

- Three additional command parameters are required as <char2>, <char3>, and <char4>.
- The use of Port 3 or Port 4 require a 4-Port instrument.
- The second additional parameter <char2> represents the numerator. The numerator values are selected from values of A1, A2, A, A4, B1, B2, B3, B4, or 1 (one).
- The third additional parameter <char3> represents the denominator and is also selected from values of A1, A2, A, A4, B1, B2, B3, B4, or 1.
- The fourth additional parameter <char4> defines the power source port and is selected from PORT1, PORT2, PORT3, PORT4, L1, or L2.
- For example, the command :CALC1:PAR1:DEF USR,A1,B1,PORT1 sets a user-defined S-Parameter value of A1/B1 for Port1.
- The USR parameter is provided for users who want to deviate from standard S-Parameter measurements. A typical application of the USR parameter would be to remove external effects from antenna analysis where a test antenna is compared to a reference antenna by using a ratio such as B2/A2.
- The command syntax for user-defined S-Parameters:

  ```
  :CALCulate{1-16}:PARameter{1-16}:DEFine <char1>,<char2>,
  <char3>,<char4>
  ```

Query Outputs

The query outputs vary depending on the command parameters that were set in the command string.

Query Output for S-Parameters, Mixed-Mode Parameters, or Noise Figure Parameters

If the command sets S11, S12, S21, S22, S13, S23, S33, S31, S32, S14, S24, S34, S41, S42, S43, S44, MIXed, NFIG, NPOW, or NTEMP:

- The query outputs only the selected parameter.
- For example, with a command setting S11, the query returns the following:

```
:CALC1:PAR1:DEF S11
:CALC1:PAR1:DEF?
S11
```

Query Output for EXT1 or EXT2

If the command sets external analog inputs EXT1 or EXT2:

- The output parameters are <char1> EXT1 | EXT2 and <char2> PORT1 | PORT2 | PORT3 | PORT4 | L1 | L2
- The query outputs the external analog input value and the associated port.
- For example, the command and query below return the following:

```
:CALC1:PAR1:DEF EXT1,PORT2
:CALC1:PAR1:DEF?
EXT1,PORT2
```

Query Output for USR

If the command sets USR for a user-defined S-Parameter, the output parameters are:

- <char1> USR
- <char2> A1 | A2 | B1 | B2 | 1
- <char3> A1 | A2 | B1 | B2 | 1
- <char4> PORT1 | PORT2
- The query outputs USR, the numerator, denominator, and the associated port. Note that the numerator and denominator are separated by a forward slash (“/”).
- For example, the command and query below return the following:

```
:CALC1:PAR1:DEF USR,A1,B1,PORT1
:CALC1:PAR1:DEF?
USR,A1/B1,PORT1
```

Cmd Parameters: The required command parameters depend on the first parameter selected:

- <char1> S11 | S12 | S21 | S22 | S13 | S23 | S33 | S31 | S32 | S14 | S24 | S34 | S41 | S42 | S43 | S44 | MIXed | NFIG | NPOW | NTEMP | EXT1 | EXT2 | USR
- <char2> A1 | A2 | A3 | A4 | B1 | B2 |B31 | B4 | 1
- <char3> A1 | A2 | A3 | A4 | B1 | B2 |B31 | B4 | 1
- <char4> PORT1 | PORT2 | PORT3 | PORT4 | L1 | L2

Query Parameters: The query parameters returned depend on which S-Parameter was selected for the command:

- <char1> S11 | S12 | S21 | S22 | S13 | S23 | S33 | S31 | S32 | S14 | S24 | S34 | S41 | S42 | S43 | S44 | MIX | NFIG | NPOW | NTEMP | EXT1 | EXT2 | USR
- <char2> A1 | A2 | A3 | A4 | B1 | B2 |B31 | B4 | 1
- <char3> A1 | A2 | A3 | A4 | B1 | B2 |B31 | B4 | 1
- <char4> PORT1 | PORT2 | PORT3 | PORT4 | L1 | L2

Output: <char1> | <char1>,<char2> | <char1>,<char2>/<char3>,<char4>

Range: NA

Default Value: The command and query default values depend on the which parameter was selected for <char1>:

- S11 for PAR1
- S12 for PAR2
- S21 for PAR3
- S22 for PAR4
- S11 for all others

Syntax Example:

```
:CALC1:PAR1:DEF S11
:CALC1:PAR1:DEF NFIG
:CALC1:PAR1:DEF EXT1,PORT2
:CALC1:PAR1:DEF EXT1,PORT2
:CALC1:PAR1:DEF USR A1,B1,PORT1
:CALC1:PAR1:DEF?
```

:CALCulate{1-16}:PARameter{1-16}:FORMat <char>
:CALCulate{1-16}:PARameter{1-16}:FORMat?

Description: The command selects the display format of the indicated trace on the indicated channel. See Chapter 2, “Programming the VectorStar Series VNA” for more information. The available display types are:

- GDELay = Group Delay
- IMAGinar = Imaginary
- LINPHase = Linear Mag and Phase
- LOGPHase = Log Magnitude and Phase
- MLINear = Linear Magnitude
- MLOGarithmic = Log Magnitude
- PHASe = Phase
- PLINear = Linear Polar and Linear Phase
- PLINCOMPlex = Linear Polar and Real/Imaginary
- PLOGarithmic = Log Polar and Log Phase
- PLOGCOMPlex = Log Polar and Real/Imaginary
- PWRIn = Power In
- PWROut = Power Out
- REAL = Real
- REIMaginary = Real and Imaginary
- SADCOMPlex = Smith (G + jB) Admittance Real/Imaginary
- SADLINear = Smith (G + jB) Admittance Linear/Phase
- SADLOGarithmic = Smith (G + jB) Admittance Log/Phase
- SADMittance = Smith (G + jB) Admittance
- SADMLC = Smith (G + jB) Admittance L/C
- SCOMPlex = Smith (R + jX) Impedance Real/Imaginary
- SIMPLC = Smith (R + jX) Impedance L/C
- SLINear = Smith (R + jX) Impedance Linear/Phase
- SLOGarithmic = Smith (R + jX) Impedance Log/Phase
- SMITh = Smith (R + jX) Impedance
- SWR = SWR
- ZREAL = Impedance Real
- ZCOMPlex = Impedance Real and Imaginary
- ZIMAGinary = Impedance Imaginary
- ZMAGNitude = Impedance Magnitude

The query outputs the display format of the indicated trace on the indicated channel. See Chapter 2, “Programming the VectorStar Series VNA” for more information and a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <char> GDELay | IMAGinary | LINPHase | LOGPHase | MLINear | MLOGarithmic | PHASe | PLINear | PLINCOMPlex | PLOGarithmic | PLOGCOMPlex | PWRIn | PWROut | REAL | REIMaginary | SADCOMPlex | SADLINear | SADLOGarithmic | SADMittance | SADMLC | SCOMPlex | SIMPLC | SLINear | SLOGarithmic | SMITh | SWR | ZREAL | ZCOMPlex | ZIMAGinary | ZMAGNitude

Query Parameters: <char> GDEL | IMAG | LINPH | LOGPH | MLIN | MLOG | PHAS | PLIN | PLINCOMP | PLOG | PLOGCOMP | PWRI | PWRO | REAL | REIM | SADCOMP | SADLIN | SADLOG | SADM | SADMLC | SCOMP | SIMPLC | SLIN | SLOG | SMIT | SWR | ZREAL | ZCOMP | ZIMAG | ZMAGN

Range: NA

Default Value: The command/query default values depend on the value of the PARameter keyword as it varies from PAR1 to PAR16 listed below:

- SMITh for PAR1 (PARameter1)
- LOGPHase for PAR2
- LOGPHase for PAR3
- SMITh for PAR4
- MLOG for PAR5, PAR6, PAR7, PAR8, PAR9, PAR10, PAR11, PAR12, PAR13, PAR14, PAR15, and PAR16.

Syntax Example: `:CALC1:PAR1:FORM GDE`

`:CALC1:PAR1:FORM?`

:CALCulate{1-16}:PARameter{1-16}:MLOCation <char>
:CALCulate{1-16}:PARameter{1-16}:MLOCation?

Description: The command sets the location of the marker readout at the given trace of the indicated channel. Command parameter definitions are:

- ULEFt = Upper Left
- URIGht = Upper Right
- LLEFt = Lower Left
- LRIGht = Lower Right
- DSOFf = Display Off

The query outputs the location of the marker readout at the given trace of the indicated channel.

Cmd Parameters: <char> ULEFt | URIGht | LLEFt | LRIGht | DSOFf

Query Parameters: NA

Query Output: <char> ULEF|URIG|LLEF|LRIG|DSOF

Range: NA

Default: ULEF

Syntax Example: `:CALC1:PAR:MLOC URIG`

`:CALC1:PAR:MLOC?`

:CALCulate{1-16}:PARameter:SELect?

Description: Query only. The query outputs the number of the active trace number on the indicated channel. Use :CALCulate{1-16}:PARameter{1-16}:SELect to select the active trace.

Cmd Parameters: NA

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 16

Default Value: 1

Syntax Example: `:CALC2:PAR:SEL?`

5-21 :CALCulate{1-16}:PARameter{1-16}:REFerece Subsystem

The :CALCulate{1-16}:PARameter:REFerence subsystem commands configure the types of parameters used for the calibration reference plane.

:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:AUTOmatic

Description: Calculate and apply the Reference Plane extension delay for the indicated trace on the indicated channel. No query.

Cmd Parameters: NA

Query Parameters: NA

Output: NA

Range: NA

Default: NA

Syntax Example: `:CALC1:PAR1:REF:EXT:AUTO`

:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:DISTance <NRf>
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:DISTance?

Description: Sets the Reference Plane extension distance in meters for the indicated trace on the indicated channel. The query outputs the Reference Plane extension distance in meters for the indicated trace on the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: NA

Default: NA

Syntax Example: `:CALC1:PAR1:REF:EXT:DIST <NRf>`

`:CALC1:PAR1:REF:EXT:DIST?`

:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:LOSS <NRf>
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:LOSS?

Description: Sets the Reference Plane extension loss in dB for the indicated trace on the indicated channel. The query outputs the Reference Plane extension loss in dB for the indicated trace on the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: NA

Default: NA

Syntax Example: `:CALC1:PAR1:REF:EXT:LOSS <NRf>`

`:CALC1:PAR1:REF:EXT:LOSS?`

:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:PHAse <NRf>
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:PHAse?

Description: Sets the Reference Plane extension phase offset in degrees for the indicated trace on the indicated channel. The query outputs the Reference Plane extension phase offset in degrees for the indicated trace on the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: NA

Default: NA

Syntax Example: `:CALC1:PAR1:REF:EXT:PHA <NRf>`

`:CALC1:PAR1:REF:EXT:PHA?`

:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:TIMe <NRf>
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:TIMe?

Description: Sets the Reference Plane extension time in seconds for the indicated trace on the indicated channel. The query outputs the Reference Plane extension time in seconds for the indicated trace on the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: NA

Default: NA

Syntax Example: `:CALC1:PAR1:REF:EXT:TIM <NRf>`

`:CALC1:PAR1:REF:EXT:TIM?`

5-22 :CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem

The :CALCulate{1-16}PARameter{1-16}:FSIMulator commands select a mapped-to port pair for 2-Port VNA measurements. Use these commands to configure the ports for a multiport system measuring a DUT with various port pairs and none, one, or two singletons.

Measurement and Mixed Mode Subsystems

- ":CALCulate{1-16}:MXP Subsystem" on page 5-77
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:DEFine <char>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:DEFine?

Description: For 2-Port VNAs only. The command sets the Mixed Mode measurement parameter for the D1S0 Differential Device for the indicated channel and trace. The query outputs the Mixed Mode measurement parameter for the D1S0 Differential Device for the indicated channel and trace.

Cmd Parameters: <char> SDD | SDC | SCD | SCC

Where the parameters are defined as Reception at Pair 1 and Drive at Pair 1 as:

- SDD = S-Parameter for differential reception at Pair 1 and differential drive at Pair 1
- SDC = S-Parameter for differential reception at Pair 1 and common-mode drive at Pair 2
- SCD = S-Parameter for common-mode reception at Pair 1 and differential drive at Pair 1
- SCC = S-Parameter for common-mode reception at Pair 1 and common-mode drive at Pair 1

Query Parameters: <char> SDD | SDC | SCD | SCC

Range: NA

Default: SDD

Syntax Example: `:CALC1:PAR1:FSIM:BAL:D1S0:DEF SCD`

`:CALC1:PAR1:FSIM:BAL:D1S0:DEF?`

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:TOPology <char>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:TOPology?

Description: This command for 2-Port or 4-Port VNAs. The command sets the D1S0 Device logical ports to the actual ports mapping as one differential pair and no singletons for the indicated channel and trace. The query outputs the D1S0 Device logical port to actual ports mapping for the indicated channel and trace.

Cmd Parameters: <char> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

Query Parameters: <char> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

Range: NA

Default: MAP12

Syntax Example: `:CALC1:PAR1:FSIM:BAL:D1S0:TOP MAP32`

`:CALC1:PAR1:FSIM:BAL:D1S0:TOP?`

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:DEFine <char>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:DEFine?

Description: The command sets the Mixed Mode measurement parameter for the D1S1 Differential Device for the indicated channel and trace. This command requires a Four-Port test set. The query outputs the Mixed Mode measurement parameter for the D1S1 Differential Device for the indicated channel and trace.

Cmd Parameters: <char> SXX | SXD | SXC | SDX | SCX | SDD | SDC | SCD | SCC

Where the mixed-mode configurations are:

- SXX = S-Parameter for first singleton reception and first singleton drive
- SXD = S-Parameter for first singleton reception and differential drive at Pair 1
- SXC = S-Parameter for first singleton reception and common-mode drive at Pair 1
- SDX = S-Parameter for differential reception at Pair 1 and first singleton drive
- SCX = S-Parameter for common-mode reception at Pair 1 and first singleton drive
- SDD = S-Parameter for differential reception at Pair 1 and differential drive at Pair 1
- SDC = S-Parameter for differential reception at Pair 1 and common-mode drive at Pair 2
- SCD = S-Parameter for common-mode reception at Pair 1 and differential drive at Pair 1
- SCC = S-Parameter for common-mode reception at Pair 1 and common-mode drive at Pair 1

Query Parameters: <char> SXX | SXD | SXC | SDX | SCX | SDD | SDC | SCD | SCC

Range: NA

Default: SXX

Syntax Example: `:CALC1:PAR1:FSIM:BAL:D1S1:DEF SXX`

`:CALC1:PAR1:FSIM:BAL:D1S1:DEF?`

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:TOPology <char1>, <char2>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:TOPology?

Description: The command sets the D1S1 Differential Device logical ports to actual ports mapping for the indicated channel and trace. Note that port mappings must be port exclusive. The query outputs the D1S1 Differential Device logical ports to actual ports mapping for the indicated channel and trace.

Cmd Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

<char2> MAP1 | MAP2 | MAP3 | MAP4

Query Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

<char2> MAP1 | MAP2 | MAP3 | MAP4 | NONE

Range: NA

Default: MAP12,MAP3

Syntax Example: `:CALC1:PAR1:FSIM:BAL:D1S1:TOP MAP34, MAP2`

`:CALC1:PAR1:FSIM:BAL:D1S1:TOP?`

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:DEFine <char>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:DEFine?

Description: The command sets the Mixed Mode measurement parameter for the D1S2 Differential Device as one differential pair and two singletons for the indicated channel and trace. This command requires a Four-Port test set. The query outputs the Mixed Mode measurement parameter for the D1S2 Differential Device for the indicated channel and trace.

Cmd Parameters: <char> SXX | SXY | SYX | SYY | SXD | SXC | SYD | SYC | SDX | SDY | SCX | SCY | SDD | SDC | SCD | SCC

Where the mixed-mode configurations are:

- SXX = S-Parameter for first singleton reception and first singleton drive
- SXY = S-Parameter for first singleton reception and second singleton drive
- SYX = S-Parameter for second singleton reception and first singleton drive
- SYY = S-Parameter for second singleton reception and second singleton drive
- SXD = S-Parameter for first singleton reception and differential drive at Pair 1
- SXC = S-Parameter for first singleton reception and common-mode drive at Pair 1
- SYD = S-Parameter for second singleton reception and differential drive at Pair 1
- SYC = S-Parameter for second singleton reception and common-mode drive at Pair 1
- SDX = S-Parameter for differential reception at Pair 1 and first singleton drive
- SDY = S-Parameter for differential reception at Pair 1 and second singleton drive
- SCX = S-Parameter for common-mode reception at Pair 1 and first singleton drive
- SCY = S-Parameter for common-mode reception at Pair 1 and second singleton drive
- SDD = S-Parameter for differential reception at Pair 1 and differential drive at Pair 1
- SDC = S-Parameter for differential reception at Pair 1 and common-mode drive at Pair 2
- SCD = S-Parameter for common-mode reception at Pair 1 and differential drive at Pair 1
- SCC = S-Parameter for common-mode reception at Pair 1 and common-mode drive at Pair 1

Query Parameters: <char> Return characters are the same as command parameters.

Range: NA

Default: SXX

Syntax Example: `:CALC1:PAR1:FSIM:BAL:D1S2:DEF SYD`

`:CALC1:PAR1:FSIM:BAL:D1S2:DEF?`

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:TOPology <char1>, <char2>, <char3>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:TOPology?

Description: The command sets the D1S2 Device logical ports to actual ports mapping for the indicated channel and trace. Note that port mappings must be port exclusive. The query outputs the D1S2 Device logical ports to actual ports mapping for the indicated channel and trace.

Cmd Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

<char2> MAP1 | MAP2 | MAP3 | MAP4

<char3> MAP1 | MAP2 | MAP3 | MAP4

Query Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

<char2> MAP1 | MAP2 | MAP3 | MAP4 | NONE

<char3> MAP1 | MAP2 | MAP3 | MAP4 | NONE

Default: MAP12, MAP3, MAP4

Syntax Example: `:CALC1:PAR1:FSIM:BAL:D1S2:TOP MAP34, MAP2, MAP1`

`:CALC1:PAR1:FSIM:BAL:D1S2:TOP?`

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:DEFine <char>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:DEFine?

Description: The command sets the Mixed Mode measurement parameter for the D2S0 Differential Device as two differential pairs and no singletons for the indicated channel and trace. This command requires a 4-Port test set.

The query outputs the Mixed Mode measurement parameter for the D2S0 Differential Device for the indicated channel and trace.

Cmd Parameters: <char> SD1D1 | SD1D2 | SD2D1 | SD2D2 | SC1C1 | SC1C2 | SC2C1 | SC2C2 | SD1C1 | SD1C2 | SD2C1 | SD2C2 | SC1D1 | SC1D2 | SC2D1 | SC2D2

Where the following balun types are available:

- SD1D1 = S-parameter for differential reception at Pair 1 and differential drive at Pair 2
- SD1D2 = S-parameter for differential reception at Pair 1 and differential drive at Pair 2
- SD2D1 = S-parameter for differential reception at Pair 2 and differential drive at Pair 1
- SD2D2 = S-parameter for differential reception at Pair 2 and differential drive at Pair 2
- SC1D1 = S-parameter for common-mode reception at Pair 1 and differential drive at Pair 1
- SC1D2 = S-parameter for common-mode reception at Pair 1 and differential drive at Pair 2
- SC2D1 = S-parameter for common-mode reception at Pair 2 and differential drive at Pair 1
- SC2D2 = S-parameter for common-mode reception at Pair 2 and differential drive at Pair 2
- SD1C1 = S-parameter for differential reception at Pair 1 and common-mode drive at Pair 2
- SD1C2 = S-parameter for differential reception at Pair 1 and common-mode drive at Pair 2
- SD2C1 = S-parameter for differential reception at Pair 2 and common-mode drive at Pair 1
- SD2C2 = S-parameter for differential reception at Pair 2 and common-mode drive at Pair 2
- SC1C1 = S-parameter for common-mode reception at Pair 1 and common-mode drive at Pair 1
- SC1C2 = S-parameter for common-mode reception at Pair 1 and common-mode drive at Pair 2
- SC2C1 = S-parameter for common-mode reception at Pair 2 and common-mode drive at Pair 1
- SC2C2 = S-parameter for common-mode reception at Pair 2 and common-mode drive at Pair 2

Query Parameters: <char> Return characters are the same as command parameters.

Range: NA

Default: SD1D1

Syntax Example:

```
:CALC1:PAR1:FSIM:BAL:D2S0:DEF SD1D1
:CALC1:PAR1:FSIM:BAL:D2S0:DEF?
```

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:TOPology <char1>, <char2>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:TOPology?

Description: The command sets the D2S0 Device logical ports to actual ports mapping for the indicated channel and trace. Note that port mappings must be port exclusive. The query outputs the D2S0 Device logical ports to actual ports mapping for the indicated channel and trace.

Cmd Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

<char2> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43

Query Parameters: <char1> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

<char2> MAP12 | MAP21 | MAP13 | MAP31 | MAP23 | MAP32 | MAP14 | MAP41 | MAP24 | MAP42 | MAP34 | MAP43 | NONE

Range: NA

Default: MAP12,MAP34

Syntax Example: `:CALC1:PAR1:FSIM:BAL:D2S0:TOP MAP13, MAP24`

`:CALC1:PAR1:FSIM:BAL:D2S0:TOP?`

:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:DEVice <char>
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:DEVice?

Description: The command selects the Balance Device type for the Balance Simulator function for the indicated channel and trace where the device type can be:

- D2S0 = Two differential pairs and no singletons.
- D1S1 = One differential pair and one singleton.
- D1S2 = One differential pair and two singletons.
- D1S0 = One differential pair and no singletons.

The query outputs the Balance Device type for the Balance Simulator function for the indicated channel and trace.

Cmd Parameters: <char> D2S0 | D1S1 | D1S2 | D1S0

Query Parameters: <char> D2S0 | D1S1 | D1S2 | D1S0

Range: NA

Default: D1S1 with a Four-Port test set.

D1S0 without a Four-Port test set.

Syntax Example: `:CALC1:PAR1:FSIM:BAL:DEV D2S0`

`:CALC1:PAR1:FSIM:BAL:DEV?`

5-23 :CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem

The :CALCulate{1-16}:PARameter{1-16}:MARKer subsystem commands control the active marker, the discrete marker mode, and whether the markers are displayed.

Marker Subsystems

Related marker configuration, control, and reporting commands are described in the following subsystems:

- ":CALCulate{1-16}:DISPlay:MARKer Subsystem" on page 5-17
- ":CALCulate{1-16}:MARKer Subsystem" on page 5-76
- ":CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem" on page 5-107
- ":CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem" on page 5-110
- ":CALCulate{1-16}[:SELected]:MARKer Subsystem" on page 5-165
- ":CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem" on page 5-178
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}:PARameter{1-16}:MARKer:ACTivate?

Description: Query only. The query outputs the number of the active marker on the indicated channel and trace. If there is no active marker, the query returns a 0 (zero). Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

Cmd Parameters: NA

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 13

Default Value: 0

Syntax Example: `:CALC1:PAR5:MARK1:ACT`

`:CALC1:PAR5:MARK:ACT?`

:CALCulate{1-16}:PARameter{1-16}:MARKer:DISCrete <char>
:CALCulate{1-16}:PARameter{1-16}:MARKer:DISCrete?

Description: The command turns the discrete marker mode on/off on the given trace of the indicated channel. The query outputs the discrete marker mode on/off state on the given trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:PAR1:MARK:DISC ON`

`:CALC1:PAR1:MARK:DISC?`

:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:ACTivate

Description: The command makes the indicated marker the active marker on the indicated channel and trace. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker. No query.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PAR1:MARK1:ACT`

:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:X <NRf>
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:X?

Description: The command enters the frequency, distance, or time value of the indicated marker and turns the marker on. The example below supposes that a point exists at 10.0E6 and is within the predefined frequency range. The query outputs the frequency, distance, or time value of the indicated marker. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

Cmd Parameters: <NRf> The input parameter is in Hertz, Meters, or Seconds.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

Range: The range is user-defined by a start and stop frequency, a start and stop distance, or a start and stop time.

Default Value: For the X-axis, the default is the Minimum Instrument Frequency.

Syntax Example: `:CALC1:PAR1:MARK1:X 10.0E6`

`:CALC1:PAR1:MARK1:X?`

:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:Y?

Description: Query only. The query outputs the response value of the indicated marker. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker. Use the following command to set the display type:

`:CALCulate{1-16}[:SELected]:FORMat`

See Chapter 2, “Programming the VectorStar Series VNA” for more information and a complete listing of trace graph types, default settings, and available ranges.

Query Parameters: <NR3> | <NR3>,<NR3> The output parameter depends on the display type.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PAR1:MARK1:Y?`

:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}[:STATe] <char>
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}[:STATe]?

Description: For the indicated channel and trace, the command toggles the indicated marker display on/off. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

For the indicated channel and the indicated trace, the query outputs the marker display on/off status of the indicated marker. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PAR1:MARK1 ON`

`:CALC1:PAR1:MARK1?`

5-24 :CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem

The :CALCulate{1-16}:PARameter{1-16}:MSTatistics subsystem commands control the marker data table display and output its values.

Marker Subsystems

Related marker configuration, control, and reporting commands are described in the following subsystems:

- ":CALCulate{1-16}:DISPlay:MARKer Subsystem" on page 5-17
- ":CALCulate{1-16}:MARKer Subsystem" on page 5-76
- ":CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem" on page 5-107
- ":CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem" on page 5-110
- ":CALCulate{1-16}[:SELected]:MARKer Subsystem" on page 5-165
- ":CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem" on page 5-178
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}:PARameter{1-16}:MSTatistics <char>
:CALCulate{1-16}:PARameter{1-16}:MSTatistics?

Description: For the indicated channel and trace, the command toggles the marker statistics display on and off. Markers do not need to be turned on to display the statistics data. If the statistics are toggled on, the statistics are determined for the entire sweep range. The three values displayed are Mean (Mean), Standard Deviation (Std Dev), and Peak-to-Peak (P-P), each measured in dB. Markers do not need to be displayed to output the statistics. If the marker statistics are displayed, a label of Stat Range Entire Sweep (Statistics Range - Entire Sweep) appears above the statistic values. The query, for the indicated channel and trace, outputs the marker statistics display on/off state.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PAR1:MST ON`

`:CALC1:PAR1:MST?`

:CALCulate{1-16}:PARameter{1-16}:MSTatistics:DATA?

Description: Query only. On the indicated channel, for the indicated trace, the query outputs the three marker statistics of the upper display. The statistics are determined for the entire sweep range. The three values output are Mean (Mean), Standard Deviation (Std Dev), and Peak-to-Peak (P-P), each measured in dB. The values are measured for the entire sweep range for the indicated trace. Markers do not need to be on to output (or display) the statistics.

Cmd Parameters: NA

Query Parameters: <NR3>, <NR3>, <NR3> Outputs marker on-screen statistics in units of dB.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PAR1:MST:DATA?`

:CALCulate{1-16}:PARameter{1-16}:MSTatistics:DATA2?

Description: Query only. Outputs the marker statistics of the lower display on the given trace of the indicated channel. A dual display trace must be set up before sending the query. The statistics are determined for the entire sweep range. The three values output are Mean (Mean), Standard Deviation (Std Dev), and Peak-to-Peak (P-P), each measured in dB. The values are measured for the entire sweep range for the indicated trace. Markers do not need to be on to output (or display) the statistics.

Cmd Parameters: NA

Query Parameters: <NR3>, <NR3>, <NR3> Outputs marker on-screen statistics.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PAR1:MST:DATA2?`

5-25 :CALCulate{1-16}:PARameter{1-16}:SELect Subsystem

The :CALCulate{1-16}:PARameter{1-16}:SELect subsystem command sets the active trace.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}:PARameter{1-16}:SELect

Description: On the indicated channel, sets the active trace. No query. To determine the active trace in the indicated channel use:

`:CALCulate{1-16}:PARameter:SELect?`

Cmd Parameters: NA

Range: NA

Default Value: 1

Syntax Example: `:CALC1:PAR1:SEL`

5-26 :CALCulate{1-16}:POLar Subsystem

The :CALCulate{1-16}:POLar subsystem commands configure the polar chart trace displays on a per-channel basis.d and how they are displayed on each trace.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}:POLar:ANGLe:STARt <NRf>
:CALCulate{1-16}:POLar:ANGLe:STARt?

Description: The command sets the start angle to use on all polar chart displays on the indicated channel.

The query outputs the start angle to use on all polar chart displays on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Degrees.

Query Parameters: <NR3> The output parameter is in Degrees.

Range: -3.6E2 to 3.6E2

Default Value: 0.00000000000E+000

Syntax Example:
```
:CALC1:POL:ANGL:STAR 2.10E1
:CALC1:POL:ANGL:STAR?
```

:CALCulate{1-16}:POLar:ANGLe:STOP <NRf>
:CALCulate{1-16}:POLar:ANGLe:STOP?

Description: The command sets the stop angle to use on all polar chart displays on the indicated channel. The query outputs the stop angle to use on all polar chart displays on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Degrees.

Query Parameters: <NR3> The output parameter is in Degrees.

Range: -3.6E2 to 3.6E2

Default Value: 3.60000000000E+002

Syntax Example: `:CALC1:POL:ANGL:STOP 2.10E1`

`:CALC1:POL:ANGL:STOP?`

:CALCulate{1-16}:POLar:CHARt <char>
:CALCulate{1-16}:POLar:CHARt?

Description: The command sets the polar chart mode of all polar chart displays on the indicated channel. The query outputs the polar chart mode of all polar chart displays on the indicated channel.

Cmd Parameters: <char> MAGPhase | MAGSweep

Query Parameters: <char> MAGP | MAGS

Range: NA

Default Value: MAGP

Syntax Example: `:CALC1:POL:CHAR MAGS`

`:CALC1:POL:CHAR?`

5-27 :CALCulate{1-16}:PROCessing:ORDer Subsystem

The :CALCulate{1-16}:PROCessing:ORDer subsystem configures the measurement port-processing order for the reference plane and group delay.

Time Domain, Group Delay, and Reference Plane Subsystems

Related time domain, group delay, and reference plane subsystems are:

- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:GCOMpression Subsystem" on page 5-149
- ":CALCulate{1-16}[:SELected]:GDELay Subsystem" on page 5-152
- ":CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem" on page 5-199
- ":SENSe{1-16}:CORRection:EXTension Subsystem" on page 5-395

:CALCulate{1-16}:PROCessing:ORDer:GRPDelay <char>
:CALCulate{1-16}:PROCessing:ORDer:GRPDelay?

Description: The command sets the order of processing of group delay and trace math data.

The query outputs the order of processing of group delay and trace math data.

Cmd Parameters: <char> GRPDtracemath | TRACemathgrpd

Query Parameters: <char> GRPD | TRAC

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PROC:ORD:GRPD TRAC`

`:CALC1:PROC:ORD:GRPD?`

:CALCulate{1-16}:PROCessing:ORDer:REFPlane <char>
:CALCulate{1-16}:PROCessing:ORDer:REFPlane?

Description: The command sets the order of processing of reference plane and impedance transformation data.

The query outputs the order of processing of reference plane and impedance transformation data.

Cmd Parameters: <char> IMPedrefplane | REFplaneimped

Query Parameters: <char> IMP | REF

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PROC:ORD:REFP IMP`

`:CALC1:PROC:ORD:REFP?`

5-28 :CALCulate{1-16}:PULSe Subsystem

The CALCulate{1-16}:PULSe subsystem commands control the pulse measurement and active pulse generator configurations. Requires the PulseView™ option to be installed.

Pulse Configuration and Setup Validation

The pulse setup commands in this subsystem require a sequence of steps and a valid setup:

1. Before sending any settings, the `:CALCulate{1-16}:PULSe:CONFig:DEFine[:STATe]` command must be set to `ON`.
2. After sending settings, the `:CALCulate{1-16}:PULSe:CONFig:DEFine[:STATe]` command must be set to `OFF`.
3. Pulse period settings (PRF/PRI) must be larger than the sum of all enabled pulse widths and delays. Use `:CALCulate{1-16}:PULSe:CONFig:DEFine:DONe:INFo?` to query for pulse setup errors.

:CALCulate{1-16}:PULSe:CONFig:DEFine[:STATe] <char>
:CALCulate{1-16}:PULSe:CONFig:DEFine[:STATe]?

Description: Turns the pulse configuration define mode on/off on the given channel. The query outputs the on/off state of the pulse configuration define mode on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PULS:CONF:DEF ON`

`:CALC1:PULS:CONF:DEF?`

:CALCulate{1-16}:PULSe:CONFig:DEFine:DONe:INFo?

Description: Queries the pulse configuration for setup errors. This command may be sent after setting `:CALCulate{1-16}:PULSe:CONFig:DEFine[:STATe]` to OFF. Query only.

Cmd Parameters: NA

Query Parameters: <string> The output parameter is any combination of numbers and letters. See definition of “<string>” on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PULS:CONF:DEF:DON:INF?`

:CALCulate{1-16}:PULSe:CONTinuous:CACType <char>
:CALCulate{1-16}:PULSe:CONTinuous:CACType?

Description: Sets the Continuous Acquisition Control Type (CACT) for the given channel. Applies only to Continuous Point in Pulse and Continuous Profiling pulse modes. (See ":CALCulate{1-16}:PULSe:MODe <char>".)

Output is the continuous acquisition control type for the indicated channel

Cmd Parameters: <char> SYNC | TIMe

Query Parameters: NA

Query Output: <char> SYNC | TIM

Range: NA

Default Value: SYNC

Syntax Example: `:CALC3:PULS:CONT:CACT SYNC`

`:CALC3:PULS:CONT:CACT?`

:CALCulate{1-16}:PULSe:CONTinuous:CCDuration?

Description: Query only. Outputs the Calculated Capture Duration for the given channel. Applies only to Continuous Point in Pulse and Continuous Profiling pulse modes. (See ":CALCulate{1-16}:PULSe:MODe <char>".)

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR3> Output is in seconds

Range: NA

Default Value: 1.02e–1

Syntax Example: `:CALC1:PULS:CONT:CCD?`

:CALCulate{1-16}:PULSe:CONTinuous:DCDuration <NRf>
:CALCulate{1-16}:PULSe:CONTinuous:DCDuration?

Description: Sets the Desired Capture Duration (DCD) for continuous acquisition for the given channel. Applies only to Continuous Point in Pulse and Continuous Profiling pulse modes. (See ":CALCulate{1-16}:PULSe:MODe <char>".)

Outputs the current Desired Capture Duration for the given channel

Cmd Parameters: <NRf> The input parameter is in seconds

Query Parameters: NA

Query Output: <NR3> The output parameter is in seconds

Range: Dependent on the PRI of the current channel.

Maximum can range from 500 ms to 14 seconds.

Minimum is 0 seconds

Default Value: 200 ms

Syntax Example: `:CALC1:PULS:DCD .03`

`:CALC1:PULS:DCD?`

:CALCulate{1-16}:PULSe:CONTinuous:UDCDuration[:STATe] <char>
:CALCulate{1-16}:PULSe:CONTinuous:UDCDuration[:STATe]?

Description: Sets whether the system will use the Desired Capture Duration or the Calculated Capture Duration. Applies only to Continuous Point in Pulse and Continuous Profiling pulse modes. (See ":CALCulate{1-16}:PULSe:MODe <char>".)

Outputs whether the system is using the Desired Capture Duration. If the channel is not in Continuous Point-in-Pulse or Continuous Profiling, the query will return false.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PULS:CONT:UDCD ON`

`:CALC1:PULS:CONT:UDCD?`

:CALCulate{1-16}:PULSe:CONTinuous:PPOints <NRf>
:CALCulate{1-16}:PULSe:CONTinuous:PPOints?

Description: Sets the number of profiling points for Continuous Profiling for the given channel. Applies only to Continuous Profiling pulse mode. (See ":CALCulate{1-16}:PULSe:MODe <char>".)

The query outputs the number of profiling points for Continuous Profiling for the given channel.

Cmd Parameters: <NR1> The input is in points

Query Parameters: NA

Query Output: <NR1> The output is in points:

Range: 2 – 25000

Default Value: 51

Syntax Example: `:CALC1:PULS:CONT:PPO 201`

`:CALC1:PULS:CONT:PPO?`

:CALCulate{1-16}:PULSe:COUPle:RECeiver:PARameter[:STATe] <char>
:CALCulate{1-16}:PULSe:COUPle:RECeiver:PARameter[:STATe]?

Description: Turns the receiver parameter coupling on/off on the given channel. Outputs the on/off state of the receiver parameter coupling on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:PULS:COUP:REC:PAR ON`

`:CALC1:PULS:COUP:REC:PAR?`

:CALCulate{1-16}:PULSe:EXTernal:SYNC:MARKing[:STATe]
:CALCulate{1-16}:PULSe:EXTernal:SYNC:MARKing[:STATe]?

Description: Turns External Sync Marking on/off on the given channel. Outputs the External Sync Marking on/off state of the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PULS:EXT:SYNC:MARK ON`

`:CALC1:PULS:EXT:SYNC:MARK?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:DELay <NRf>
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:DELay?

Description: Sets the burst delay on an indicated generator and channel. Output is the burst delay on the indicated generator and channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 0 s to 0.25 s in 2.5 ns increments

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:PULS:GEN1:BURS:DEL 1E-1`

`:CALC1:PULS:GEN1:BURS:DEL?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:NUMBer <NRf>
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:NUMBer?

Description: Sets the number of bursts on the indicated generator and channel. Output is the number of bursts on the indicated generator and channel.

Cmd Parameters: <NRf> The number of bursts is a unitless number.

Query Parameters: <NR1> The returned number of bursts is an integer.

Range: 1 to 65535

Default Value: 1

Syntax Example: `:CALC1:PULS:GEN1:BURS:NUMB 5`

`:CALC1:PULS:GEN1:BURS:NUMB?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:PERiod <NRf>
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:PERiod?

Description: Sets the burst period on an indicated generator and channel. Output is the burst period on an indicated generator and channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 10 ns to 0.25 s in 2.5 ns increments

Default Value: 2.00000000000E-006

Syntax Example: `:CALC1:PULS:GEN1:BURS:PER 4E-6`

`:CALC1:PULS:GEN1:BURS:PER?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:WIDTh <NRf>
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:WIDTh?

Description: Sets the burst width on the indicated generator and channel. Output is the burst width on the indicated generator and channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 5 ns to 0.25 s in 2.5 ns increments

Default Value: 1.00000000000E-006

Syntax Example: `:CALC1:PULS:GEN1:BURS:WIDT 2E-6`

`:CALC1:PULS:GEN1:BURS:WIDT?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:DUTy:CYCLe?

Description: Outputs the indicated pulse generator duty cycle on the indicated channel.

Query Parameters: <NR3> The output parameter is in percent.

Range: 0 to 100

Default Value: 1.00000000000E+001

Syntax Example: `:CALC1:PULS:GEN1:DUT:CYCL?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:ENABle <char>
:CALCulate{1-16}:PULSe:GENerator{1-4}:ENABle?

Description: Sets the pulse generator enable status on the indicated generator and channel. Output is the pulse generator enable status on the indicated generator and channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PULS:GEN1:ENAB ON`

`:CALC1:PULS:GEN1:ENAB?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:INVerted:POLarity <char>
:CALCulate{1-16}:PULSe:GENerator{1-4}:INVerted:POLarity?

Description: Sets the pulse generator inverted polarity on the indicated generator and channel. Output is the pulse generator inverted polarity on the indicated generator and channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PULS:GEN1:INV:POL ON`

`:CALC1:PULS:GEN1:INV:POL?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:LABel <string>
:CALCulate{1-16}:PULSe:GENerator{1-4}:LABel?

Description: Sets the pulse generator label on the indicated generator and channel. Output is the pulse generator label on the indicated generator and channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters. See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters. See definition of “<char>” on page 2-14.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:PULS:GEN1:LAB 'IC7000'`

`:CALC1:PULS:GEN1:LAB?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:MODe <char>
:CALCulate{1-16}:PULSe:GENerator{1-4}:MODe?

Description: Sets the pulse generator mode on the indicated channel. Output is the pulse generator mode on the indicated channel.

Cmd Parameters: <char> SINGlet | DOUBlet | TRIPlet | QUADruplet | BURSt

Query Parameters: <char> SING | DOUB | TRIP | QUAD | BURS

Range: NA

Default Value: SING

Syntax Example: `:CALC1:PULS:GEN1:MOD SING`

`:CALC1:PULS:GEN1:MOD?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:DELay <NRf>
:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:DELay?

Description: Sets the pulse delay on the indicated generator and channel. Output is the pulse delay on the indicated generator and channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 0 s to 0.25 s in 2.5 ns increments

Default Value: Pulse 1: 0.00000000000E+000
Pulse 2: 2.00000000000E-006
Pulse 3: 4.00000000000E-006
Pulse 4: 6.00000000000E-006

Syntax Example: `:CALC1:PULS:GEN1:PULS1:DEL 1E-6`

`:CALC1:PULS:GEN1:PULS1:DEL?`

:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:WIDTh <NRf>
:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:WIDTh?

Description: Sets the pulse width on the indicated generator and channel. Output is the pulse width on the indicated generator and channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 5 ns to 0.25 s in 2.5 ns increments

Default Value: Pulse 1: 1.00000000000E-006
Pulse 2: 1.00000000000E-006
Pulse 3: 1.00000000000E-006
Pulse 4: 1.00000000000E-006

Syntax Example: `:CALC1:PULS:GEN1:PULS1:WIDT 2E-6`

`:CALC1:PULS:GEN1:PULS1:WIDT?`

:CALCulate{1-16}:PULSe:MEASure:GENenerator
:CALCulate{1-16}:PULSe:MEASure:GENenerator?

Description: Sets the pulse generators on the given channel to be enabled only during a measurement. Outputs the pulse generators enabled only during measurement status on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Output: <char> 1 | 0

Default Value: 0

Syntax Example: `:CALC1:PULS:MEAS:GEN 1`

`:CALC1:PULS:MEAS:GEN`

:CALCulate{1-16}:PULSe:MEASure:WIDTh <NRf>
:CALCulate{1-16}:PULSe:MEASure:WIDTh?

Description: Sets the pulse measurement width on an indicated channel. Output is the pulse measurement width on an indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 2.5 ns to 1$*$ Resolution $*$ 10^8 s in 2.5 ns increments

Default Value: 1.00000000000E-006

Syntax Example: `:CALC1:PULS:MEAS:WIDT 2E-6`

`:CALC1:PULS:MEAS:WIDT?`

:CALCulate{1-16}:PULSe:MODe <char>
:CALCulate{1-16}:PULSe:MODe?

Description: Sets the pulse measurement mode on the indicated channel. Query output is the pulse measurement mode on the indicated channel. Pulse mode definitions are:

PROFile = Pulse Profile

PIPulse = Point in Pulse

P2Pulse = Pulse to Pulse

CPIPulse = Continuous Point in Pulse

CPRofile = Continuous Profile

Cmd Parameters: <char> PIPulse | P2PulSe | PROFile | CPIPulse | CPRofile

Query Parameters: NA

Query Output: <char> PIP | P2P | PROF | CPIP | CPR

Range: NA

Default Value: PIP

Syntax Example: `:CALC1:PULS:MOD PROF`

`:CALC1:PULS:MOD?`

:CALCulate{1-16}:PULSe:POINts <NRf>
:CALCulate{1-16}:PULSe:POINts?

Description: Sets the number of points on the indicated channel. Output is the number of points on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: When :CALCulate{1-16}:PULSe:MODe is set to PIPulse: 1
When :CALCulate{1-16}:PULSe:MODe is set to P2Pulse and :CALCulate{1-16}:PULSe:SYNC:TYPe is set to CONTinuous: 1
Otherwise: 1 to 25,000 or 1 to 100,000

Default Value: When :CALCulate{1-16}:PULSe:MODe is set to PIPulse: 1
When :CALCulate{1-16}:PULSe:MODe is set to P2Pulse and :CALCulate{1-16}:PULSe:SYNC:TYPe is set to CONTinuous: 1
Otherwise: 51

Syntax Example: `:CALC1:PULS:POIN 5.12E2`

`:CALC1:PULS:POIN?`

:CALCulate{1-16}:PULSe:PRF <NRf>
:CALCulate{1-16}:PULSe:PRF?

Description: Sets the pulse PRF on an indicated channel. Output is the pulse PRF on an indicated channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 1/MaxPRI to 1/MinPRI

Default Value: 1.00000000000E+005

Syntax Example: `:CALC1:PULS:PRF 2E5`

`:CALC1:PULS:PRF?`

:CALCulate{1-16}:PULSe:PRI <NRf>
:CALCulate{1-16}:PULSe:PRI?

Description: Sets the pulse PRI on an indicated channel. Output is the pulse PRI on an indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 20 ns to Resolution $* 10^8$ s in 2.5 ns increments

Default Value: 1.00000000000E-005

Syntax Example: `:CALC1:PULS:PRI 4E-5`

`:CALC1:PULS:PRI?`

:CALCulate{1-16}:PULSe:PULSes <NRf>
:CALCulate{1-16}:PULSe:PULSes?

Description: Sets the number of pulses on the given channel. Output is the number of pulses on the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: When :CALCulate{1-16}:PULSe:MODe is set to PIPulse: 1
When :CALCulate{1-16}:PULSe:MODe is set to P2Pulse and :CALCulate{1-16}:PULSe:SYNC:TYPe is set to CONTinuous: 1
Otherwise: 1 to 25,000 or 1 to 100,000

Default Value: When :CALCulate{1-16}:PULSe:MODe is set to PIPulse: 1
When :CALCulate{1-16}:PULSe:MODe is set to P2Pulse and :CALCulate{1-16}:PULSe:SYNC:TYPe is set to CONTinuous: 1
Otherwise: 2.50000000000E+004

Syntax Example: `:CALC1:PULS:PULS 5E3`

`:CALC1:PULS:PULS?`

:CALCulate{1-16}:PULSe:RECeiver:TYPe <char>
:CALCulate{1-16}:PULSe:RECeiver:TYPe?

Description: Sets the pulse receiver type on the indicated channel. Output is the pulse receiver type on the indicated channel.

Cmd Parameters: <char> A1|A2|A3|A4|B1|B2|B3|B4

Query Parameters: <char> A1|A2|A3|A4|B1|B2|B3|B4

Range: NA

Default Value: B2

Syntax Example: `:CALC1:PULS:REC:TYP B1`

`:CALC1:PULS:REC:TYP?`

:CALCulate{1-16}:PULSe:RESolution <NRf>
:CALCulate{1-16}:PULSe:RESolution?

Description: Command sets the resolution of the pulse acquisition of the indicated channel.

Query outputs the pulse resolution on the indicated channel

Cmd Parameters: <NRf> The command parameter is in seconds

Query Output: <NR3> Query output is in seconds

Range: 2.5 ns to 70 ns

Default Value: 2.525 ns

Syntax Example: `:CALC1:PULS:RES 10E-9`

`:CALC1:PULS:RES?`

:CALCulate{1-16}:PULSe:STARt:DELay <NRf>
:CALCulate{1-16}:PULSe:STARt:DELay?

Description: Sets the pulse measurement start delay on an indicated channel. Output is the pulse measurement start delay on an indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 0 s to 2 $*$ Resolution $*$ 10^8 s in 2.5 ns increments

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:PULS:STAR:DEL 1E-1`

`:CALC1:PULS:STAR:DEL?`

:CALCulate{1-16}:PULSe:STOP:DELay <NRf>
:CALCulate{1-16}:PULSe:STOP:DELay?

Description: Sets the pulse measurement stop delay on an indicated channel. Output is the pulse measurement stop delay on an indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 0 s to 2* Resolution * 10^8 s in 2.5 ns increments

Default Value: 2.00000000000E-006

Syntax Example: `:CALC1:PULS:STOP:DEL 4E-6`

`:CALC1:PULS:STOP:DEL?`

:CALCulate{1-16}:PULSe:SYNC:TYPe <char>
:CALCulate{1-16}:PULSe:SYNC:TYPe?

Description: Sets the pulse synch type on the indicated channel. Outputs the pulse synch type on the indicated channel.

Cmd Parameters: <char> INTernal|EXTFalling|EXTRising | CONTinuous

Query Parameters: <char> INT|EXTF|EXTR | CONT

Range: NA

Default Value: INT

Syntax Example: `:CALC1:PULS:SYNC:TYP EXTR`

`:CALC1:PULS:SYNC:TYP?`

:CALCulate{1-16}:PULSe:ZOOM:MARKer[:STATe] <char>
:CALCulate{1-16}:PULSe:ZOOM:MARKer[:STATe]?

Description: Sets the on/off state of the pulse zoom marker on the given channel. Output is the on/off state of the pulse zoom marker on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:PULS:ZOOM:MARK ON`

`:CALC1:PULS:ZOOM:MARK?`

5-29 :CALCulate{1-16}:REFerence Subsystem

The :CALCulate{1-16}:REFerence subsystem commands configure various parameters related to the reference plane in line types such as coaxial, microstrip, and waveguides.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

Time Domain, Group Delay, and Reference Plane Subsystems

Related time domain, group delay, and reference place subsystems are:

- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:GCOMpression Subsystem" on page 5-149
- ":CALCulate{1-16}[:SELected]:GDELay Subsystem" on page 5-152
- ":CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem" on page 5-199
- ":SENSe{1-16}:CORRection:EXTension Subsystem" on page 5-395

:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric <char>
:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric?

Description: The command sets the reference plane extension coaxial line dielectric type on the indicated channel.

The query outputs the reference plane extension coaxial line dielectric type on the indicated channel.

Cmd Parameters: <char> AIR | MICROporous | OTHER | POLYethylene | TEFLON

Query Parameters: <char> AIR | MICRO | OTHER | POLY | TEFLON

Range: NA

Default Value: AIR

Syntax Example: `:CALC1:REF:EXT:COAX:DIEL AIR`

`:CALC1:REF:EXT:COAX:DIEL?`

:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric:OTHer <NRf>
:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric:OTHer?

Description: The command sets the reference plane extension coaxial line manual dielectric value on the indicated channel.

The query outputs the reference plane extension coaxial line manual dielectric value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: 1 to 9.99E3

Default Value: 1.00000000000E+000

Syntax Example: `:CALC1:REF:EXT:COAX:DIEL:OTH 1.0E3`

`:CALC1:REF:EXT:COAX:DIEL:OTH?`

:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric:VALue?

Description: Query only.

The query outputs the reference plane extension coaxial line dielectric value on the indicated channel.

Cmd Parameters: NA

Query Parameters: <NR3> The output parameter is a unitless number.

Range: NA

Default Value: 1.00064900000E+000

Syntax Example: `:CALC1:REF:EXT:COAX:DIEL:VAL?`

:CALCulate{1-16}:REFerence:EXTension:LINE <char>
:CALCulate{1-16}:REFerence:EXTension:LINE?

Description: The command sets the reference plane extension line type on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters. The query outputs the reference plane extension line type on the indicated channel.

Cmd Parameters: <char> COAXial | MICROstrip | NONDISpersive | WAVEguide

Query Parameters: <char> COAX | MICRO | NONDIS | WAVE

Range: NA

Default Value: COAX

Syntax Example: `:CALC1:REF:EXT:LINE COAX`

`:CALC1:REF:EXT:LINE?`

:CALCulate{1-16}:REFerence:EXTension:MICrostrip:DIELectric <NRf>
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:DIELectric?

Description: The command sets the reference plane extension microstrip substrate dielectric value on the indicated channel. The query outputs the reference plane extension microstrip substrate dielectric value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: Command range: 1 to 10
Query range: 1 to 9.99E3

Default Value: 9.96000000000E+000

Syntax Example: `:CALC1:REF:EXT:MIC:DIEL 2.80`

`:CALC1:REF:EXT:MIC:DIEL?`

:CALCulate{1-16}:REFerence:EXTension:MICrostrip:EFFective <NRf>
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:EFFective?

Description: The command sets the reference plane extension microstrip effective dielectric value on the indicated channel. The query outputs the reference plan extension microstrip effective dielectric value on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Cmd Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: 6.69000000000E+000

Syntax Example: `:CALC1:REF:EXT:MIC:EFF 2.80`

`:CALC1:REF:EXT:MIC:EFF?`

:CALCulate{1-16}:REFerence:EXTension:MICrostrip:THICKness <NRf>
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:THICKness?

Description: The command sets the reference plane extension microstrip substrate thickness on the indicated channel. The query outputs the reference plane extension microstrip substrate thickness on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 2.54000000000E-004

Syntax Example: `:CALC1:REF:EXT:MIC:THICK 3.0E-4`

`:CALC1:REF:EXT:MIC:THICK?`

:CALCulate{1-16}:REFerence:EXTension:MICrostrip:WIDth <NRf>
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:WIDth?

Description: The command sets the reference plane extension microstrip width on the indicated channel. The query outputs the reference plane extension microstrip width on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 2.38760000000E-004

Syntax Example: `:CALC1:REF:EXT:MIC:WID 3.0E-4`

`:CALC1:REF:EXT:MIC:WID?`

:CALCulate{1-16}:REFerence:EXTension:MICrostrip:Z0 <NRf>
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:Z0?

Description: The command sets the reference plane extension microstrip impedance on the indicated channel. The query outputs the reference plane extension microstrip impedance on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: 5.00000000000E+001

Syntax Example: `:CALC1:REF:EXT:MIC:Z0 7.5E1`

`:CALC1:REF:EXT:MIC:Z0?`

:CALCulate{1-16}:REFerence:EXTension:PARameter <char>
:CALCulate{1-16}:REFerence:EXTension:PARameter?

Description: Selects the reference extension parameter type Port or Trace. Outputs the reference extension parameter type Port or Trace.

Cmd Parameters: <char> PORT | TRACe

Query Parameters: NA

Output: <char> PORT|TRAC

Range: NA

Default: NA

Syntax Example: `:CALC1:REF:EXT:PAR <char>`

`:CALC1:REF:EXT:PAR?`

:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:AUTOmatic

Description: The command calculates and applies the reference plane extension delay for the indicated port on the indicated channel. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:REF:EXT:PORT1:AUTO`

:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:DISTance <NRf>
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:DISTance?

Description: The command sets the reference plane extension distance in meters for the indicated port on the indicated channel. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. The query outputs the reference plane extension distance in meters for the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:REF:EXT:PORT1:DIST 5E-4`

`:CALC1:REF:EXT:PORT1:DIST?`

:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:LOSS <NRf>
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:LOSS?

Description: The command sets the reference plane extension loss in dB for the indicated port on the indicated channel. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. The query outputs the reference plane extension loss in dB for the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: -1E3 to +1E3

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:REF:EXT:PORT1:LOSS 3E0`

`:CALC1:REF:EXT:PORT1:LOSS?`

:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:PHAse <NRf>
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:PHAse?

Description: The command sets the reference plane extension phase offset in degrees for the indicated port on the indicated channel. The allowable range is -360 degrees to +360 degrees. If the -360 value is exceeded, the instrument truncates the input to -360; if the +360 value is exceeded, the instrument truncates the input to +360. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. The query outputs the reference plane extension phase offset in degrees for the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Degrees.

Query Parameters: <NR3> The output parameter is in Degrees.

Range: -360 to +360

Default Value: 0.000000E+000

Syntax Example: `:CALC1:REF:EXT:PORT1:PHA 1.5E1`

`:CALC1:REF:EXT:PORT1:PHA?`

:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:TIMe <NRf>
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:TIMe?

Description: The command sets the reference plane extension in time in seconds for the indicated port on the indicated channel. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. The query outputs the reference plane extension in time in seconds for the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:REF:EXT:PORT1:TIM 5.0E-2`

`:CALC1:REF:EXT:PORT1:TIM?`

:CALCulate{1-16}:REFerence:EXTension:WAVeguide:DIELectric <NRf>
:CALCulate{1-16}:REFerence:EXTension:WAVeguide:DIELectric?

Description: The command sets the reference plane extension waveguide dielectric value on the indicated channel. The query outputs the reference plane extension waveguide dielectric value on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: 1.00000000000E+000

Syntax Example: `:CALC1:REF:EXT:WAV:DIEL 2.5E0`

`:CALC1:REF:EXT:WAV:DIEL?`

:CALCulate{1-16}:REFerence:EXTension:WAVeguide:FREQuency <NRf>
:CALCulate{1-16}:REFerence:EXTension:WAVeguide:FREQuency?

Description: The command sets the reference plane extension waveguide cutoff frequency on the indicated channel. The query outputs the reference plane extension waveguide cutoff frequency on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example:
```
:CALC1:REF:EXT:WAV:FREQ 1E10
:CALC1:REF:EXT:WAV:FREQ?
```

5-30 :CALCulate{1-16}[:SELected]:CONVersion Subsystem

The :CALCulate{1-16}[:SELected]:CONVersion subsystem sets the parameter conversion configuration and control for the indicated channel and the active trace.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:CONVersion:FUNCtion <char>
:CALCulate{1-16}[:SELected]:CONVersion:FUNCtion?

Description: The command sets the parameter conversion function on the active trace of the indicated channel. The query outputs the parameter conversion function on the active trace of the indicated channel.

Cmd Parameters: <char> ZREFlection | ZTRansmit | YREFlection | YTRansmit | INVersion

Where:

- ZREFlection = Sets the conversion parameter to Z:Reflection.
- ZTRansmit = Sets the conversion parameter to Z:Transmission.
- YREFlection = Sets the conversion parameter to Y:Reflection.
- YTRansmit = Sets the conversion parameter to Y:Transmission.
- INVersion = Inverts the parameter to 1/S.

Query Parameters: <char> ZREF | ZTR | YREF | YTR | INV

Range: NA

Default Value: INV

Syntax Example:

```
:CALC1:CONV:FUNC ZREF
:CALC1:CONV:FUNC?
```

:CALCulate{1-16}[:SELected]:CONVersion[:STATe] <char>
:CALCulate{1-16}[:SELected]:CONVersion[:STATe]?

Description: The command sets the on/off status of parameter conversion on the active trace of the indicated channel. The query outputs the on/off status of parameter conversion on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Default Value: 0

Syntax Example: `:CALC1:CONV ON`

`:CALC1:CONV?`

5-31 :CALCulate{1-16}[:SELected]:DATA Subsystem

The :CALCulate{1-16}[:SELected]:DATA subsystem commands input and output various instrument information sets such as trace data and S-Parameters.

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- ":CALCulate{1-16}:FORMat Subsystem - SnP Data" on page 5-49
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":FORMat Subsystem" on page 5-242
- ":HCOPy Subsystem" on page 5-245
- ":MMEMory Subsystem" on page 5-251

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:DATA:FDATa <block>
:CALCulate{1-16}[:SELected]:DATA:FDATa?

Description: The command inputs formatted data to display on the active trace. The instrument must be in Hold Sweep. The query outputs formatted data of the active trace.

Cmd Parameters: <block> See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <block> See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:DATA:FDAT <block>`

`:CALC1:DATA:FDAT?`

:CALCulate{1-16}[:SELected]:DATA:FMEMory <block>
:CALCulate{1-16}[:SELected]:DATA:FMEMory?

Description: The command inputs formatted data to display on the active trace and writes it to trace memory. The command can also perform math functions. The query outputs the formatted memory of the active trace.

Cmd Parameters: <block> See definition of "<block> or <arbitrary block>" on page 2-12.

Cmd Parameters: <block> See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:DATA:FMEM <block>`

`:CALC1:DATA:FMEM?`

:CALCulate{1-16}[:SELected]:DATA:SDATa <Block>
:CALCulate{1-16}[:SELected]:DATA:SDATa?

Description: The command inputs corrected real and imaginary S-parameter data to display on the active trace for the indicated channel. The data must be real and imaginary formatted data. The command converts the active trace display appropriately. The query outputs the real/imaginary S-parameter data for the active trace on the indicated channel.

Cmd Parameters: <Block> See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <Block>

Output: <Block>

Range: NA

Default Value: NA

Syntax Example: `:CALC1:DATA:SDAT <block>`

`:CALC1:DATA:SDAT?`

:CALCulate{1-16}[:SELected]:DATA:SMEMory <Block>
:CALCulate{1-16}[:SELected]:DATA:SMEMory?

Description: The command inputs corrected S-parameter trace memory to display on the active trace. The query outputs corrected S-parameter trace memory of the active trace.

Cmd Parameters: <block> See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <block> See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:DATA:SMEM <block>`

`:CALC1:DATA:SMEM?`

5-32 :CALCulate{1-16}[:SELected]:EYE Subsystem

The :CALCulate{1-16}[:SELected]:EYE subsystem commands set configuration parameters for EYE calculations for the indicated channel.

:CALCulate{1-16}[:SELected]:EYE:EXECute

Description: Command performs the calculation of the eye diagram for the active trace of the given channel.

Cmd Parameters: NA

Query Parameters: NA

Query Output: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EYE:EXEC`

:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:TYPE <char>
:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:TYPE?

Description: Command sets the input bit pattern type on the active trace of the given channel.

Query Outputs the bit pattern type on the active trace of the given channel.

Parameter definitions are:

- `PRBS` = Pseudo-Random Bit Sequence
- `K285` = K 28.5
- `USER` = User fixed pattern

Cmd Parameters: <char> PRBS | K285 | USER

Query Parameters: NA

Query Output: <char> PRBS | K285 | USER

Range: NA

Default Value: PRBS

Syntax Example: `:CALC1:EYE:INPut:BPATtern:TYPE K285`

`:CALC1:EYE:INPut:BPATtern:TYPE?`

:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:LENGth <int>
:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:LENGth?

Description: The command sets bits' power of 2 for PRBS / K28.5 pattern. This value is used only when the bit pattern type is selected as PRBS / K28.5. (In the user fixed pattern scenario, the pattern length is set by the number of bits entered by the user (up to a max limit of 64).)

Query outputs the bit pattern length on the active trace of the given channel.

Cmd Parameters: <int> 5 | 7 | 9 | 11 | 13 | 15 | 17 | 19 | 21

Query Parameters: NA

Query Output: <int>

31 = $2^5 - 1$

127= $2^7 - 1$

511= $2^9 - 1$

2047= $2^{11} - 1$

8191= $2^{13} - 1$

32767= $2^{15} - 1$

131071= $2^{17} - 1$

524287= $2^{19} - 1$

2097151= $2^{21} - 1$

Range: NA

Default Value: 511

Syntax Example: `:CALC1:EYE:INP:BPAT:LENG 11`

`:CALC1:EYE:INP:BPAT:LENG?`

:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:USER <string>

Description: Sets the user fixed pattern on the active trace of the given channel.l.

Cmd Parameters: < string > String with 0s and 1s only, max length allowed is 64bits.

Query Parameters: NA

Query Output: NA

Range: Maximum length allowed is 64 bits

Default Value: NA

Syntax Example: `:CALC1:EYE:INPut:BPATtern:USER '10001011110'`

:CALCulate{1-16}[:SELected]:EYE:INPut:DRATe <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:DRATe?

Description: The command sets the input data rate (bit rate) on the active trace of the given channel.

Query outputs the data rate (bit rate) on the active trace of the given channel.

Cmd Parameters: <nrf> data rate value in GBPS

Query Parameters: NA

Query Output: <nr3> outputs the set data rate value in GBPS

Range: Data rate should be positive numbers only

Default Value: 1 GBPS

Syntax Example: `:CALC1:EYE:INP:DRAT 2`

`:CALC1:EYE:INP:DRAT?`

:CALCulate{1-16}[:SELected]:EYE:INPut:HSHift <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:HSHift?

Description: Command sets the horizontal shift of eye on the active trace of the given channel.

Query outputs the horizontal shift of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> Horizontal shift represented in fraction of a Bit Period

Query Parameters: NA

Query Output: <nr3> outputs the horizontal shift represented in fraction of a Bit Period

Range: –0.5 to +0.5

Default Value: 0

Syntax Example: `:CALC1:EYE:INP:HSH 0.5`

`:CALC1:EYE:INP:HSH?`

:CALCulate{1-16}[:SELected]:EYE:INPut:RTIMe:DATA <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:RTIMe:DATA?

Description: Command sets the rise time value of the input bit stream on the active trace of the given channel.

Query outputs the rise time value of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> rise time value in picoseconds (ps)

Query Parameters: NA

Query Output: <nr3> outputs the rise time value in picoseconds

Range: 0 and larger, but may not be > 0.4/(Data rate)

Default Value: 0 ps

Syntax Example: `:CALC1:EYE:INP:RTIM:DATA 15`

`:CALC1:EYE:INP:RTIM:DATA?`

:CALCulate{1-16}[:SELected]:EYE:INPut:FTIMe:DATA <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:FTIMe:DATA?

Description: Command sets the fall time value of the input bit stream on the active trace of the given channel.

Query outputs the fall time value of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> fall time value in picoseconds (ps)

Query Parameters: NA

Query Output: <nr3> outputs the fall time value in picoseconds

Range: 0 and larger, but may not be > 0.4/(Data rate)

Default Value: 0 ps

Syntax Example: `:CALC1:EYE:INP:FTIM:DATA 25`

`:CALC1:EYE:INP:FTIM:DATA?`

:CALCulate{1-16}[:SELected]:EYE:INPut:HLEVel <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:HLEVel?

Description: Command sets the high voltage level on the active trace of the given channel. (high level > low level)

Query outputs the high voltage level of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> High voltage level in mV

Query Parameters: NA

Query Output: <nr3> Outputs the high voltage level in mV

Range: –5000 mV to +5000 mV

Default Value: 1000 mV

Syntax Example: `:CALC1:EYE:INP:HLEV 5000`

`:CALC1:EYE:INP:HLEV?`

:CALCulate{1-16}[:SELected]:EYE:INPut:LLEVel <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:LLEVel?

Description: Command sets the low voltage level on the active trace of the given channel. (high level > low level)

Query outputs the low voltage level of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> Low voltage level in mV

Query Parameters: NA

Query Output: <nr3> Outputs the low voltage level in mV

Range: –5000 mV to +5000 mV

Default Value: 0 mV

Syntax Example: `:CALC1:EYE:INP:LLEV 3000`

`:CALC1:EYE:INP:LLEV?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom[:STATe] <char>
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom[:STATe]?

Description: Command sets the Jitter-Random RMS on/off for the active trace of the given channel

Query outputs the Jitter-Random RMS state for the active trace of the given channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EYE:INP:JITT:RAND 1`

`:CALC1:EYE:INP:JITT:RAND?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom:MAGNitude <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom:MAGNitude?

Description: Command sets the Jitter-Random Magnitude of the input bit stream on the active trace of the given channel. This value is used only when Random RMS state is ON.

Query outputs the Jitter-Random Magnitude of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> Magnitude value in picoseconds (ps)

Query Parameters: NA

Query Output: <nr3> Outputs the magnitude value in picoseconds

Range: Must be ≥ 0 and < 0.4/(data rate).

Default Value: 0 ps

Syntax Example: `:CALC1:EYE:INP:JITT:RAND:MAGN 100`

`:CALC1:EYE:INP:JITT:RAND:MAGN?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}[:STATe] <char>
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}[:STATe]?

Description: Command sets the Jitter-Dirac 1/2 settings on/off for the active trace of the given channel.

Query outputs the Jitter-Dirac 1/2 settings state for the active trace of the given channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EYE:INP:JITT:DIR1 1`

`:CALC1:EYE:INP:JITT:DIR1?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:OFFSet <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:OFFSet?

Description: Command sets the Jitter-Dirac 1/2 offset of the input bit stream on the active trace of the given channel. This value is used only when Dirac 1 settings state is ON.

Query outputs the Jitter-Dirac 1/2 offset of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> Offset value in picoseconds (ps)

Query Parameters: NA

Query Output: <nr1> Outputs the offset value in picoseconds

Range: Must be ≥ 0 and < 0.4/(data rate).

Default Value: 0 ps

Syntax Example: `:CALC1:EYE:INP:JITT:DIR1:OFFS 10`

`:CALC1:EYE:INP:JITT:DIR1:OFFS?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:PROBability <NRF> :CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:PROBability?

Description: Command sets the Jitter-Dirac 1/2 probability of the input bit stream on the active trace of the given channel. This value is used only when Dirac 1/2 settings state is ON.

Query outputs the Jitter-Dirac 1/2 probability of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nr2> probability value

Query Parameters: NA

Query Output: <nr2> outputs the probability value

Range: ≥ 0 and ≤ 1 (The sum of the two probabilities cannot exceed 1 if both are enabled.)

Default Value: 0.1

Syntax Example: `:CALC1:EYE:INP:JITT:DIR1:PROB 0.2`

`:CALC1:EYE:INP:JITT:DIR1:PROB?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}[:STATe] <char> :CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}[:STATe]?

Description: Command sets the Jitter-Sinusoidal 1 /2 settings on/off for the active trace of the given channel.

Query outputs the Jitter-Sinusoidal 1 /2 settings state for the active trace of the given channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EYE:INP:JITT:SIN1 1`

`:CALC1:EYE:INP:JITT:SIN1?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:AMPLitude <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:AMPLitude ?

Description: Command sets the Jitter-Sinusoidal 1/2 amplitude of the input bit stream on the active trace of the given channel. This value is used only when Sinusoidal 1 /2 settings state is ON.

Query outputs the Jitter-Sinusoidal 1/2 amplitude of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> Amplitude value in picoseconds

Query Parameters: NA

Query Output: <nr1> Outputs the amplitude value in picoseconds

Range: Must be ≥ 0 and < 0.4/(data rate).

Default Value: 0 ps

Syntax Example: `:CALC1:EYE:INP:JITT:SIN1:AMPL 10`

`:CALC1:EYE:INP:JITT:SIN1:AMPL?`

:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:FREQuency <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:FREQuency ?

Description: Command sets the Jitter-Sinusoidal 1/2 frequency of the input bit stream on the active trace of the given channel. This value is used only when Sinusoidal 1/2 settings state is ON.

Query outputs the Jitter-Sinusoidal 1/2 frequency of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> Frequency value in Hz

Query Parameters: NA

Query Output: <nr1> Outputs the frequency value in Hz

Range: Must be > 0 and < data rate.

Default Value: 1000000 Hz

Syntax Example: `:CALC1:EYE:INP:JITT:SIN1:FREQ 1000`

`:CALC1:EYE:INP:JITT:SIN1:FREQ?`

:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe[:STATe] <char>
:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe[:STATe]?

Description: Command sets the Noise-RMS Amplitude on/off for the active trace of the given channel.

Query outputs the Noise-RMS Amplitude state for the active trace of the given channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:EYE:INP:NOIS 1`

`:CALC1:EYE:INP:NOIS?`

:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe:AMPLitude <NRF>
:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe:AMPLitude?

Description: Command sets the Noise-RMS Amplitude of the input bit stream on the active trace of the given channel. This value is used only when Noise settings state is ON.

Query outputs the Noise-RMS Amplitude of the input bit stream on the active trace of the given channel.

Cmd Parameters: <nrf> amplitude value in mV

Query Parameters: NA

Query Output: <nr3> outputs the amplitude value in mV

Range: Must be less than high-low level/2

Default Value: 0 mV

Syntax Example: `:CALC1:EYE:INP:NOIS:AMPL 100`

`:CALC1:EYE:INP:NOIS:AMPL?`

:CALCulate{1-16}[:SELected]:EYE:INPut:PERSistence <NR1>
:CALCulate{1-16}[:SELected]:EYE:INPut:PERSistence?

Description: Command sets the persistence value on the active trace of the given channel.

Query outputs the persistence value on the active trace of the given channel.

Cmd Parameters: <nr1> Persistence value

Query Parameters: NA

Query Output: <nr1> Outputs the set persistence value

Range: 1 to 10000

Default Value: 200

Syntax Example: `:CALC1:EYE:INP:PERS 2`

`:CALC1:EYE:INP:PERS?`

:CALCulate{1-16}[:SELected]:EYE:RESults:DATA?

Description: Query only. Outputs the 'eye measurement results' for the active trace of the indicated channel.

Fourteen values (amplitude and time related parameters) are returned. This query will always return the values of all the amplitude and time related parameters of the selected eye trace, irrespective of whether the display for each parameter is turned ON or OFF.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <string> This query returns string in <ACSII> format consisting of 14 fields separated by commas. The fields are:

- <Level Zero: {numeric}>
- <Level One: {numeric}>
- <Level Mean: {numeric}>
- <Amplitude: {numeric}>
- <Height: {numeric}>
- <Opening factor: {numeric}>
- <SNR: {numeric}>
- <CrossPercentage: {numeric}>
- <Width: {numeric}>
- <Rise time: {numeric}>
- <Fall time: {numeric}>
- <PP jitter: {numeric}>
- <RMS jitter: {numeric}>
- <Duty cycle distortion: {numeric}>

Range: NA

Default Value: NA

Syntax Example: `:CALC1:EYE:RES:DATA?`

5-33 :CALCulate{1-16}[:SELected]:FORmat Subsystem

The :CALCulate{1-16}[:SELected]:FORmat subsystem command configures the display type for the active trace on a per-channel basis.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:FORMat <char>
:CALCulate{1-16}[:SELected]:FORMat?

Description: The command selects the display format of the active trace on the indicated channel. The query outputs the display format of the active trace on the indicated channel. The available display types are:

- GDELay = Group Delay
- IMAGinar = Imaginary
- LINPHase = Linear Mag and Phase
- LOGPHase = Log Magnitude and Phase
- MLINear = Linear Magnitude
- MLOGarithmic = Log Magnitude
- PHASe = Phase
- PLINear = Linear Polar and Linear Phase
- PLINCOMPlex = Linear Polar and Real/Imaginary
- PLOGarithmic = Log Polar and Log Phase
- PLOGCOMPlex = Log Polar and Real/Imaginary
- PWRIn = Power In
- PWROut = Power Out
- REAL = Real
- REIMaginary = Real and Imaginary
- SADCOMPlex = Smith (G + jB) Admittance Real/Imaginary
- SADLINear = Smith (G + jB) Admittance Linear/Phase
- SADLOGarithmic = Smith (G + jB) Admittance Log/Phase

- SADMittance = Smith (G + jB) Admittance
- SADMLC = Smith (G + jB) Admittance L/C
- SCOMPlex = Smith (R + jX) Impedance Real/Imaginary
- SIMPLC = Smith (R + jX) Impedance L/C
- SLINear = Smith (R + jX) Impedance Linear/Phase
- SLOGarithmic = Smith (R + jX) Impedance Log/Phase
- SMITh = Smith (R + jX) Impedance
- SWR = SWR
- ZREAL = Impedance Real
- ZCOMPlex = Impedance Real and Imaginary
- ZIMAGinary = Impedance Imaginary
- ZMAGNitude = Impedance Magnitude

Cmd Parameters: <char> GDELay | IMAGinary | LINPHase | LOGPHase | MLINear | MLOGarithmic | PHASe | PLINear | PLINCOMPlex | PLOGarithmic | PLOGCOMPlex | PWRIn | PWROut | REAL | REIMaginary | SADCOMPlex | SADLINear | SADLOGarithmic | SADMittance | SADMLC | SCOMPlex | SIMPLC | SLINear | SLOGarithmic | SMITh | SWR | ZREAL | ZCOMPlex | ZIMAGinary | ZMAGNitude

Query Parameters: <char> GDEL | IMAG | LINPH | LOGPH | MLIN | GDEL | IMAG | LINPH | LOGPH | MLIN | MLOG | PHAS | PLIN | PLINCOMP | PLOG | PLOGCOMP | PWRI | PWRO | REAL | REIM | SADCOMP | SADLIN | SADLOG | SADM | SADMLC | SCOMP | SIMPLC | SLIN | SLOG | SMIT | SWR | ZREAL | ZCOMP | ZIMAG | ZMAGN

Range: NA

Default Value: SMIT

Syntax Example: `:CALC1:FORM LOGPH`

`:CALC1:FORM?`

5-34 :CALCulate{1-16}[:SELected]:GCOMpression Subsystem

The :CALCulate{1-16}[:SELected]:GCOMpression subsystem configures and controls the gain compression instrument functions.

Time Domain, Group Delay, and Reference Plane Subsystems

Related time domain, group delay, and reference place subsystems are:

- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:GCOMpression Subsystem" on page 5-149
- ":CALCulate{1-16}[:SELected]:GDELay Subsystem" on page 5-152
- ":CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem" on page 5-199
- ":SENSe{1-16}:CORRection:EXTension Subsystem" on page 5-395

:CALCulate{1-16}[:SELected]:GCOMpression:INDicator[:STATe] <char>
:CALCulate{1-16}[:SELected]:GCOMpression:INDicator[:STATe]?

Description: The command turns on/off the compression point indicator on the active trace of the indicated channel. The instrument must be in Power Sweep CW mode. The query outputs the on/off status of the compression point indicator on the active trace of the indicated channel. The instrument must be in Power-Sweep CW mode.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:GCOM:IND ON`

`:CALC1:GCOM:IND?`

:CALCulate{1-16}[:SELected]:GCOMpression:PARameter <char>
:CALCulate{1-16}[:SELected]:GCOMpression:PARameter?

Description: The command sets the S-Parameter for gain compression on the active trace of the indicated channel. The instrument must be in Power-Sweep Swept-Frequency mode. In the indicated channel, the query outputs the S-Parameter for gain compression on the active trace. The instrument must be in Power-Sweep Swept-Frequency mode. This value is not reset by an instrument reset.

Cmd Parameters: <char> S11 | S12 | S21 | S22

Query Parameters: <char> S11 | S12 | S21 | S22

Range: NA

Default Value: NA

Syntax Example: `:CALC1:GCOM:PAR S22`

`:CALC1:GCOM:PAR?`

:CALCulate{1-16}[:SELected]:GCOMpression:PVALue <NRf>
:CALCulate{1-16}[:SELected]:GCOMpression:PVALue?

Description: The command sets the gain compression point value on the active trace of the indicated channel. Instrument must be in Power Sweep CW mode or in Power Sweep Swept Frequency mode. The query outputs the gain compression point value on the active trace of the indicated channel. Instrument must be in Power Sweep CW mode or in Power Sweep Swept Frequency mode.

Cmd Parameters: <NRf> The input parameter is in dB.

Cmd Parameters: <NR3> The output parameter is in dB.

Range: MPNF

Default Value: 1.000000E+000

Syntax Example: `:CALC1:GCOM:PVAL 2.0E0`

`:CALC1:GCOM:PVAL?`

:CALCulate{1-16}[:SELected]:GCOMpression:REFerence <char>
:CALCulate{1-16}[:SELected]:GCOMpression:REFerence?

Description: The command sets the reference type for gain compression on the active trace of the indicated channel. Instrument must be in Power Sweep CW mode or in Power Sweep Swept Frequency mode. There are three types of gain compression references that can be set:

- MAXGain = Maximum power gain at the highest point in the gain curve output.
- PHOLd = The reference value will be that for the S-Parameter such as S21 set by the `:CALCulate{1-16}[:SELected]:GCOMpression:PARameter` command.
- PINput = Input in dBm, this is the reference is value set for the selected S-Parameter which occurs when the input power is set to the value of gain compression reference value.

The query outputs the reference type for gain compression on the active trace of the indicated channel. Instrument must be in Power Sweep CW mode or in Power Sweep Swept Frequency mode.

Cmd Parameters: <char> MAXGain | PHOLd | PINput

Query Parameters: <char> MAXG | PHOL | PIN

Range: NA

Default Value: MAXG

Syntax Example: `:CALC1:GCOM:REF PHOL`

`:CALC1:GCOM:REF?`

:CALCulate{1-16}[:SELected]:GCOMpression:REFerence:VALue <NRf>
:CALCulate{1-16}[:SELected]:GCOMpression:REFerence:VALue?

Description: The command sets a reference level for gain compression on the active trace of the indicated channel. Instrument must be in Power Sweep CW mode or in Power Sweep Swept Frequency mode. The query outputs the reference level for gain compression on the active trace of the indicated channel. Instrument must be in Power Sweep CW mode or in Power Sweep Swept Frequency mode.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:GCOM:REF:VAL -2.0E1`

`:CALC1:GCOM:REF:VAL?`

:CALCulate{1-16}[:SELected]:GCOMpression:SNORmalize[:STATe] <char>
:CALCulate{1-16}[:SELected]:GCOMpression:SNORmalize[:STATe]?

Description: The command toggles the self normalization mode on/off for the active trace of the indicated channel. The instrument must be in Power Sweep CW mode. The query outputs the on/off status of the self normalization mode on the active trace of the indicated channel. The instrument must be in Power Sweep CW mode.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:GCOM:SNOR 1`

`:CALC1:GCOM:SNOR?`

5-35 :CALCulate{1-16}[:SELected]:GDELay Subsystem

The :CALCulate{1-16}[:SELected]:GDELay subsystem provides group delay aperture control.

Time Domain, Group Delay, and Reference Plane Subsystems

Related time domain, group delay, and reference place subsystems are:

- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:GCOMpression Subsystem" on page 5-149
- ":CALCulate{1-16}[:SELected]:GDELay Subsystem" on page 5-152
- ":CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem" on page 5-199
- ":SENSe{1-16}:CORRection:EXTension Subsystem" on page 5-395

:CALCulate{1-16}[:SELected]:GDELay:APERture <NRf>
:CALCulate{1-16}[:SELected]:GDELay:APERture?

Description: Sets the group delay aperture on group delay display for the indicated channel and active trace. Outputs the group delay aperture of a group delay display aperture for the indicated channel and active trace.

Cmd Parameters: <NRf> The input parameter is in Percent.

Query Parameters: <NR3> The output parameter is in Percent.

Range: 0 to 2E1

Default Value: 0.000000E+000

Syntax Example: `:CALC1:GDEL:APER 1.5E1`

`:CALC1:GDEL:APER?`

5-36 :CALCulate{1-16}[:SELected]:LIMit Subsystem

The :CALCulate{1-16}[:SELected]:LIMit subsystem provides limit line configuration and control for the active trace.

Limit Line and Segment Subsystems

Related limit line and segment configuration and control subsystems are:

- ":CALCulate{1-16}[:SELected]:LIMit Subsystem" on page 5-153
- ":DISPlay Subsystem" on page 5-223
- ":SENSe{1-16}:FSEGMent Subsystem" on page 5-401.
- ":SENSe{1-16}:FSEGMent{1-50} Subsystem" on page 5-412.
- ":SENSe{1-16}:ISEGMent Subsystem" on page 5-436.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446
- ":SENSe{1-16}:SEGMent Subsystem" on page 5-487

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:LIMit:DATA <block>
:CALCulate{1-16}[:SELected]:LIMit:DATA?

Description: The command inputs the limit line table for the active trace of the given channel. Outputs the limit line table of the active trace of the given channel.

Cmd Parameters: <block> Block data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <block> Block data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:LIM:DATA <block>`

`:CALC1:LIM:DATA?`

:CALCulate{1-16}[:SELected]:LIMit:DISPlay[:STATe] <char>
:CALCulate{1-16}[:SELected]:LIMit:DISPlay[:STATe]?

Description: The command turns limit display on/off for the active trace of the indicated channel. The query outputs the limit display on/off status for the active trace of the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:LIM:DISP ON`

`:CALC1:LIM:DISP?`

:CALCulate{1-16}[:SELected]:LIMit:FAIL?

Description: Query only. The query outputs the limit testing result for the active trace of the given channel, where:

- "0" = The limit passed.
- "1" = The limit failed.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 | 1

Default Value: 0

Syntax Example: `:CALC1:LIM:FAIL?`

:CALCulate{1-16}[:SELected]:LIMit:OFF

Description: The query turns all limits of the indicated channel off. No query.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:LIM:OFF`

:CALCulate{1-16}[:SELected]:LIMit:REPort:POINt?

Description: Query only. The query outputs the number of points failing limit testing.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to the current number of set measurement points.

Default Value: 0

Syntax Example: `:CALC1:LIM:REP:POIN?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:ADD {No argument} | {<char>} | {<char>,<NRf>,<NRf>}

Description: On the indicated channel, the command adds a limit line segment for the active trace. If the optional parameters are omitted, an empty segment is added.

No query.

Limit Lines for Rectilinear Displays

For rectilinear displays, up to 50 segment lines can be added to each trace display.

Limit Lines for Circular Polar and Smith Chart Displays

For circular displays (Smith Impedance Chart, Smith Admittance Chart, Linear Polar Graph, and Log Polar Graph), two limits can be defined as an Upper Limit and a Lower Limit. For example, the following commands set upper and lower limit lines:

- Clears all limit lines
  `:CALC1:LIM:SEGM:CLE`
- Adds an upper limit line
  `:CALC1:LIM:SEGM:ADD UPP`
- Sets the radius of the upper limit line to 1.321
  `:CALC1:LIM:SEGM:DEF 1.321`
- Adds a lower limit line
  `:CALC1:LIM:SEGM:ADD LOW`
- Sets the radius of the lower limit line to 0.395
  `:CALC1:LIM:SEGM:DEF 0.395`

Cmd Parameters: {<No argument>} | {<char>} | {<char>,<NRf>,<NRf>}

<No argument> If no argument is added to the command, the command adds an empty segment.

<char> UPPer | LOWer | POLYgon | NONe

<NRf> Start time, frequency, or distance for X1.

<NRf> Stop time, frequency, or distance for X2.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:LIM:SEGM:ADD UPP, 2.0E9, 3.0E9`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:CLEar

Description: The command clears all the limit segment definitions on the active trace of the indicated channel. No query.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:LIM:SEGM:CLE`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:COUNt?

Description: Query only. Outputs the number of limit segments defined on the active trace of the indicated channel.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 50

Default Value: 0

Syntax Example: `:CALC1:LIM:SEGM:ADD UPPER, 2.0E9, 3.0E9`

`:CALC1:LIM:SEGM:COUN?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:DEFine <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:DEFine?

Description: The command defines a limit line segment for an added segment for the active trace of the indicated channel. Output the current limit line segment for the active trace of the indicated channel.

- If only one <NRf> parameter is set, it defines the limit radius for a circular display.
- If two <NRf> parameters are used, it defines the Y1 start value and the Y2 stop value.
- If four <NRf> parameters are used, it defines the Y1 start value and the Y2 stop value for the upper trace in a dual trace display, and the Y12 (Y1sub in the GUI) start value and the Y22 (Y2sub in the GUI) stop value for the lower trace in a dual trace display.

Cmd Parameters: <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf>

Query Parameters: <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf>

Range: NA

Default Value: NA

Syntax Example: `:CALC1:LIM:SEGM:DEF 0.75`

`:CALC1:LIM:SEGM:DEF?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:RADius <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:RADius?

Description: Sets the radius value of the current circular limit line segment being defined for the active trace of the indicated channel. Outputs the radius value of the current circular limit line segment being defined for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM:ADD`

`:CALC1:LIM:SEGM:RAD 0.75`

`:CALC1:LIM:SEGM:RAD?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:TYPe <char>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:TYPe?

Description: Sets the limit line type of the current limit line segment being defined for the active trace of the indicated channel where:

- UPPer = Upper limit line
- LOWer = Lower limit line
- NONe = No limit line

Outputs the limit line type of the current limit line segment being defined for the active trace of the indicated channel.

Note: The limit line must exist before using this command. If no limit lines have been created, use the prerequisite command below to add one or more limit lines:

`:CALCulate{1-16}:LIMit:SEGMent:ADD`

This adds one (1) limit segment which permits the example below to work.

Cmd Parameters: <char> UPPer | LOWer | POLYgon | NONe

Query Parameters: NA

Output: <char> UPP | LOW | POLY | NON

Range: NA

Default Value: NON

Syntax Example: `:CALC1:LIM:SEGM:TYP POLY`

`:CALC1:LIM:SEGM:TYP?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X1 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X1?

Description: Sets the start X (X1) value of the current limit line segment being defined for the active trace of the indicated channel. Outputs start X (X1) value of the current limit line segment being defined for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM:X1 10E9`

`:CALC1:LIM:SEGM:X1?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X2 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X2?

Description: Sets the stop X value of the current limit line segment being defined for the active trace of the indicated channel. Outputs stop X value of the current limit line segment being defined for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM:X2 20E9`

`:CALC1:LIM:SEGM:X2?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y1 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y1?

Description: Sets the start Y value of the current limit line segment being defined for the active trace of the indicated channel. Outputs the start Y value of the current limit line segment being defined for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM:Y1 10`

`:CALC1:LIM:SEGM:Y1?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y12 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y12?

Description: Sets the start Y value of the bottom graph current limit line segment being defined for the active trace of the indicated channel. Outputs start Y value of the bottom graph current limit line segment being defined for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM:Y12 15`

`:CALC1:LIM:SEGM:Y12?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y2 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y2?

Description: Sets the stop Y value of the current limit line segment being defined for the active trace of the indicated channel. Outputs the stop Y value of the current limit line segment being defined for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM:Y2 20`

`:CALC1:LIM:SEGM:Y2?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y22 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y22?

Description: Sets the stop Y22 value of the bottom graph current limit line segment being defined for the active trace of the indicated channel. Outputs the stop Y22 value of the bottom graph current limit line segment being defined for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM:Y22 17`

`:CALC1:LIM:SEGM:Y22?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:DEFine <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf>} :CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:DEFine?

Description: Define the indexed limit line segment for the active trace of the indicated channel. The limit segment must exist. Output the indexed limit line segment for the active trace of the indicated channel.

Note that if a :SYSTem:PRESet is issued where the :SYSTem:PRESet:TYPe RESET has been previously set, any user-defined segmented limit lines are cleared and the query generates an error.

One <NRf> Parameter

If only one <NRf> parameter is set, it defines the limit radius for a circular display.

Two <NRf> Parameters

If two <NRf> parameters are used, it defines the Y1 start value and the Y2 stop value.

Four <NRf> Parameters

If four <NRf> parameters are used, it defines the Y1 start value and the Y2 stop value for the upper trace in a dual trace display, and the Y12 (Y1sub in the GUI) start value and the Y22 (Y2sub in the GUI) stop value for the lower trace in a dual trace display.

Cmd Parameters: <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf>

Query Parameters: <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM1:DEF 2.0000000000E010`

`:CALC1:LIM:SEGM1:DEF?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:DELete

Description: Deletes the indicated limit line segment of the active trace of the indicated channel.

No query.

Cmd Parameters: NA

Query Parameters: NA

Output: NA

Range: NA

Default Value: NA

Example: `:CALC1:LIM:SEGM1:DEL`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:RADius <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:RADius?

Description: Sets the radius value of the indicated circular limit line segment for the active trace of the indicated channel. Outputs the radius value of the indicated circular limit line segment for the active trace of the indicated channel.

The indicated circular limit line must exist. To create a limit line to edit, use:

`:CALCulate{1-16}:LIMit:SEGMent:ADD`

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEG:ADD`

`:CALC1:LIM:SEGM1:RAD 0.75`

`:CALC1:LIM:SEGM1:RAD?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:TYPe <char>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:TYPe?

Description: Sets the limit line type of the indicated limit line segment for the active trace of the indicated channel where:

- UPPer = Upper limit line
- LOWer = Lower limit line
- POLYgon = Polygon limit line
- NONe = No limit line

Outputs the limit line type of the indicated limit line segment for the active trace of the indicated channel.

Note: The limit line must exist before using this command. If no limit lines have been created, use the prerequisite command below to add one or more limit lines:

`:CALCulate{1-16}:LIMit:SEGMent:ADD`

This adds one (1) limit segment which permits the example below to work.

Cmd Parameters: <char> UPPer | LOWer | POLYgon | NONe

Query Parameters: NA

Output: <char> UPP | LOW | POLY |NON

Range: NA

Default Value: NON

Syntax Example: `:CALC1:LIM:SEGM1:TYP POLY`

`:CALC1:LIM:SEGM1:TYP?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X1 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X1?

Description: Sets the start X1 value of the indicated limit line segment for the active trace of the indicated channel. For a dual trace rectilinear display, this sets the X1 start value for both the upper and lower traces. Outputs start X1 value of the indicated limit line segment for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM1:X1 10E9`

`:CALC1:LIM:SEGM1:X1?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X2 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X2?

Description: Sets the stop X2 value of the indicated limit line segment for the active trace of the indicated channel.

For a dual trace rectilinear display, this sets the X1 start value for both the upper and lower traces.

Outputs stop X2 value of the indicated limit line segment for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM1:X2 20E9`

`:CALC1:LIM:SEGM1:X2?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y1 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y1?

Description: Sets the start Y1 value of the indicated limit line segment for the active trace of the indicated channel. For a dual trace rectilinear display, this sets the Y1 start value for only the upper trace. Outputs the start Y1 value of the indicated limit line segment for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM1:Y1 10`

`:CALC1:LIM:SEGM1:Y1?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y12 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y12?

Description: Only for use on dual rectilinear displays. If a single rectilinear display is used, the value is accepted but no change is made to the trace display.

Sets the start Y12 value of the bottom graph indicated limit line segment for the active trace of the indicated channel.

Outputs start Y12 value of the bottom graph indicated limit line segment for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM1:Y12 15`

`:CALC1:LIM:SEGM1:Y12?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y2 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y2?

Description: Sets the stop Y2 value of the indicated limit line segment for the active trace of the indicated channel.

For a dual trace rectilinear display, this sets the Y2 stop value for the upper trace only.

Outputs the stop Y2 value of the indicated limit line segment for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM1:Y2 20`

`:CALC1:LIM:SEGM1:Y2?`

:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y22 <NRf>
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y22?

Description: Only for use on dual rectilinear displays. If a single rectilinear display is used, the value is accepted but no change is made to the trace display.

Sets the stop Y22 value of the bottom graph indicated limit line segment for the active trace of the indicated channel.

Outputs the stop Y22 value of the bottom graph indicated limit line segment for the active trace of the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:LIM:SEGM1:Y22 17`

`:CALC1:LIM:SEGM1:Y22?`

:CALCulate{1-16}[:SELected]:LIMit[:STATe] <char>
:CALCulate{1-16}[:SELected]:LIMit[:STATe]?

Description: The command turns limit testing on/off for the active trace of the indicated channel. The query outputs the limit testing on/off status for the active trace of the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:LIM`

`:CALC1:LIM?`

5-37 :CALCulate{1-16}[:SELected]:MARKer Subsystem

The :CALCulate{1-16}:PARameter{1-16}:MARKer subsystem commands control the active marker display, value, and search functions.

Marker Subsystems

Related marker configuration, control, and reporting commands are described multiple subsystems:

- ":CALCulate{1-16}:DISPlay:MARKer Subsystem" on page 5-17
- ":CALCulate{1-16}:MARKer Subsystem" on page 5-76
- ":CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem" on page 5-107
- ":CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem" on page 5-110
- ":CALCulate{1-16}[:SELected]:MARKer Subsystem" on page 5-165
- ":CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem" on page 5-178
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:MARKer:ACTivate?

Description: Query only. On the indicated channel, the query outputs the active marker number of the active trace.

To activate a marker, use the command:

`:CALCulate{1-16}[:SELected]:MARKer{1-13}:ACTivate`

Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 13

Default Value: 0

Syntax Example: `:CALC1:MARK:ACT?`

:CALCulate{1-16}[:SELected]:MARKer:ALL[:STATe] <char>

Description: The command toggles all markers on/off on the active trace of the given channel.

No query.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:ALL ON`

:CALCulate{1-16}[:SELected]:MARKer:MOVe:CENTer

Description: The command moves the active marker range value to the stimulus center range on the active trace of the indicated channel.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:MOV:CENT`

:CALCulate{1-16}[:SELected]:MARKer:MOVe:REFMarker

Description: The command moves the active marker to the reference marker on the active trace of the indicated channel. The active marker cannot be a reference marker. The reference marker must be on.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:MOV:REFM`

:CALCulate{1-16}[:SELected]:MARKer:MOVe:STARt

Description: The command moves the active marker range value to the stimulus start range on the active trace of the indicated channel.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:MOV:STAR`

:CALCulate{1-16}[:SELected]:MARKer:MOVe:STOP

Description: The command moves the active marker range value to the stimulus stop range on the active trace of the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:MOV:STOP`

:CALCulate{1-16}[:SELected]:MARKer:MPSEArch

Description: The command performs a multiple peak search on the active trace of the indicated channel.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:MPSEA`

:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:EXCursion <NRf>
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:EXCursion?

Description: The command sets the marker search excursion value for multiple peak searches on the active trace of the indicated channel. The query outputs the marker search excursion value for multiple peak searches on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:MPSEA:EXC 1.8E1`

`:CALC1:MARK:MPSEA:EXC?`

:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:POLarity <char>
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:POLarity?

Description: The command sets the marker search polarity value for multiple peak searches on the active trace of the indicated channel. The query outputs the marker search polarity value for multiple peak searches on the active trace of the indicated channel.

Cmd Parameters: <char> POSitive | NEGative | BOTH

Query Parameters: <char> POS | NEG | BOTH

Range: NA

Default Value: POS

Syntax Example: `:CALC1:MARK:MPSEA:POL POS`

`:CALC1:MARK:MPSEA:POL?`

:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:THREshold <NRf>
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:THREshold?

Description: The command sets the marker search threshold value for multiple peak searches on the active trace of the indicated channel. The query outputs the marker search threshold value for multiple peak searches on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:MPSEA:THRE 1.8E1`

`:CALC1:MARK:MPSEA:THRE?`

:CALCulate{1-16}[:SELected]:MARKer:MTSEArch

Description: Performs a multiple target search on the active trace of the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:MTSEA`

:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TARget <NRf>
:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TARget?

Description: The command sets the marker search target value for multiple target searches on the active trace of the indicated channel. The query outputs the marker search target value for multiple target searches on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3>, <NR3>, <NR3> The output parameter depends on the display type.

Range: MPNI

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:MTSEA:TAR 1E7`

`:CALC1:MARK:MTSEA:TAR?`

:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TRANsition <char>
:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TRANsition?

Description: The command sets the marker search transition value for multiple target searches on the active trace of the indicated channel. The query outputs the marker search transition value for multiple target searches on the active trace of the indicated channel.

Cmd Parameters: <char> POSitive | NEGative | BOTH

Query Parameters: <char> POS | NEG | BOTH

Range: NA

Default Value: BOTH

Syntax Example: `:CALC1:MARK:MTSEA:TRAN POS`

`:CALC1:MARK:MTSEA:TRAN?`

:CALCulate{1-16}[:SELected]:MARKer:OFF

Description: The command turns all markers of the indicated channel off. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:OFF`

:CALCulate{1-16}[:SELected]:MARKer:PSEArch <char>
:CALCulate{1-16}[:SELected]:MARKer:PSEArch?

Description: The command performs a peak search on the active trace of the indicated channel. The query outputs the last marker search type for peak searches on the active trace of the indicated channel.

Cmd Parameters: <char> PEAK | LEFT | RIGHT

Query Parameters: <char> PEAK | LEFT | RIGHT

Range: NA

Default Value: PEAK

Syntax Example: `:CALC1:MARK:PSEA PEAK`

`:CALC1:MARK:PSEA?`

:CALCulate{1-16}[:SELected]:MARKer:PSEArch:EXCursion <NRf>
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:EXCursion?

Description: The command sets the marker search excursion value for peak searches on the active trace of the indicated channel. The query outputs the marker search excursion value for peak searches on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: MPNI

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:PSEA:EXC 1.8E1`

`:CALC1:MARK:PSEA:EXC?`

:CALCulate{1-16}[:SELected]:MARKer:PSEArch:POLarity <char>
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:POLarity?

Description: The command sets the marker search polarity value for peak searches on the active trace of the indicated channel. The query outputs the marker search polarity value for peak searches on the active trace of the indicated channel.

Cmd Parameters: <char> POSitive | NEGative | BOTH

Query Parameters: <char> POS | NEG | BOTH

Range: NA

Default Value: POS

Syntax Example: `:CALC1:MARK:PSEA:POL POS`

`:CALC1:MARK:PSEA:POL?`

:CALCulate{1-16}[:SELected]:MARKer:PSEArch:THREshold <NRf>
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:THREshold?

Description: The command sets the marker search threshold value for peak searches on the active trace of the indicated channel. The query outputs the marker search threshold value for peak searches on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:PSEA:THRE 1.8E1`

`:CALC1:MARK:PSEA:THRE?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch <char>
:CALCulate{1-16}[:SELected]:MARKer:SEArch?

Description: The command sets the marker search type for the active marker on the active trace of the indicated channel. The query outputs the marker search type for the active marker on the active trace of the indicated channel.

Cmd Parameters: <char> MAX | MIN | PEAK | TARGet

Query Parameters: <char> MAX | MIN | PEAK | TARG

Range: NA

Default Value: MAX

Syntax Example: `:CALC1:MARK:SEA MAX`

`:CALC1:MARK:SEA?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:DATA?

Description: Query only. The query outputs the marker search bandwidth data of the active marker on the active trace of the indicated channel.

Query Parameters: <NR3>, <NR3>, <NR3>, <NR3>{,<NR3>} The command outputs four or five <NR3> data elements in the following sequence:

- First <NR3> - Bandwidth in Hertz.
- Second <NR3> - Center of marker search range in Hertz, Meters, or Seconds.
- Third <NR3> - Q which is a unitless number.
- Fourth <NR3> - Loss in dB.
- Fifth <NR3> - Optional. The Shape Factor which is a unitless number. This parameter is optional depending on if the Shape Function is turned on or off using the Shape Command.

Syntax Example: `:CALC1:MARK:SEA:BAND:DATA?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:DEFine <NRf>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:DEFine?

Description: The command sets the marker search define value for bandwidth calculation of the active marker on the active trace of the indicated channel. The query outputs the marker search define value for bandwidth calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:SEA:BAND:DEF 3.0E4`

`:CALC1:MARK:SEA:BAND:DEF?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:HIGH <NRf> :CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:HIGH?

Description: The command sets the marker search high value for bandwidth shape factor calculation of the active marker on the active trace of the indicated channel. The query outputs the marker search high value for bandwidth shape factor calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:SEA:BAND:SHAP:HIGH 4E0`

`:CALC1:MARK:SEA:BAND:SHAP:HIGH?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:LOW <NRf> :CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:LOW?

Description: The command sets the marker search low value for bandwidth shape factor calculation of the active marker on the active trace of the indicated channel. The query outputs the marker search low value for bandwidth shape factor calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:SEA:BAND:SHAP:LOW 4E0`

`:CALC1:MARK:SEA:BAND:SHAP:LOW?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe[:STATe] <char> :CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe[:STATe]?

Description: The command toggles on/off the marker search bandwidth shape factor calculation of the active marker on the active trace of the indicated channel. The query outputs the on/off status of marker search bandwidth shape factor of the active marker calculation on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:SEA:BAND:SHAP 1`

`:CALC1:MARK:SEA:BAND:SHAP?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth[:STATe] <char>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth[:STATe]?

Description: The command toggles on/off the marker search bandwidth calculation of the active marker on the active trace of the indicated channel. The query outputs the on/off status of marker search bandwidth calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:SEA:BAND ON`

`:CALC1:MARK:SEA:BAND?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:DATA?

Description: Query only. The query outputs the marker search notch bandwidth data of the active marker on the active trace of the indicated channel.

Query Parameters: <NR3>, <NR3>, <NR3>, <NR3>{, <NR3>} The output parameter is in Hertz, Meters, or Seconds.

Range: Outputs multiple NR3 data

Default Value: NA

Syntax Example: `:CALC1:MARK:SEA:NOTC:DATA?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:DEFine <NRf>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:DEFine?

Description: The command sets the marker search define value for notch bandwidth calculation of the active marker on the active trace of the indicated channel. The query outputs the marker search define value for notch bandwidth calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:SEA:NOTC:DEF 1.0E4`

`:CALC1:MARK:SEA:NOTC:DEF?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:HIGH <NRf>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:HIGH?

Description: The command sets the marker search high value for notch shape factor calculation of the active marker on the active trace of the indicated channel. The query outputs the marker search high value for notch shape factor calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:SEA:NOTC:SHAP:HIGH 4E0`

`:CALC1:MARK:SEA:NOTC:SHAP:HIGH?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:LOW <NRf>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:LOW?

Description: The command sets the marker search low value for notch shape factor calculation of the active marker on the active trace of the indicated channel. The query outputs the marker search low value for notch shape factor calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:SEA:NOTC:SHAP:LOW 4E0`

`:CALC1:MARK:SEA:NOTC:SHAP:LOW?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe[:STATe] <char>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe[:STATe]?

Description: The command toggles on/off the marker search notch shape factor calculation of the active marker on the active trace of the indicated channel. Outputs on/off status of marker search notch shape factor calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:SEA:NOTC:SHAP ON`

`:CALC1:MARK:SEA:NOTC:SHAP?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh[:STATe] <char>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh[:STATe]?

Description: The command toggles on/off the marker search notch calculation of the active marker on the active trace of the indicated channel. The query outputs the on/off status of marker search notch calculation of the active marker on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:SEA:NOTC ON`

`:CALC1:MARK:SEA:NOTC?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:ALL[:STATe] <char>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:ALL[:STATe]?

Description: The command toggles on/off applying the marker search range to all traces of the indicated channel. The query outputs the on/off status of applying the marker search range to all traces of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:SEA:RANG:ALL ON`

`:CALC1:MARK:SEA:RANG:ALL?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STARt:X <NRf>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STARt:X?

Description: The command sets the marker search range start range value on the active trace of the indicated channel. The query outputs the marker search range start range value on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Hertz, Meters, or Seconds.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:MARK:SEA:RANG:STAR:X 2.5E9`

`:CALC1:MARK:SEA:RANG:STAR:X?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STOP:X <NRf>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STOP:X?

Description: The command sets the marker search range stop range value on the active trace of the indicated channel. The query outputs the marker search range stop range value on the active trace of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Hertz, Meters, or Seconds.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

]Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:MARK:SEA:RANG:STOP:X 2.5E9`

`:CALC1:MARK:SEA:RANG:STOP:X?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe[:STATe] <char>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe[:STATe]?

Description: The command toggles on/off the marker search range on the active trace of the indicated channel. The query outputs the on/off status of marker search range on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:SEA:RANG ON`

`:CALC1:MARK:SEA:RANG?`

:CALCulate{1-16}[:SELected]:MARKer:SEArch:TRACKing[:STATe] <char>
:CALCulate{1-16}[:SELected]:MARKer:SEArch:TRACKing[:STATe]?

Description: The command toggles on/off marker search tracking on the active trace of the indicated channel. The query outputs the on/off status of marker search tracking on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK:SEA:TRACK ON`

`:CALC1:MARK:SEA:TRACK?`

:CALCulate{1-16}[:SELected]:MARKer:SET:CENTer

Description: The command sets the stimulus center range to the active marker range value on the active trace of the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:SET:CENT`

:CALCulate{1-16}[:SELected]:MARKer:SET:REFLevel

Description: The command sets the display reference level to the active marker response value on the active trace of the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:SET:REFL`

:CALCulate{1-16}[:SELected]:MARKer:SET:STARt

Description: The command sets the stimulus start range to the active marker range value on the active trace of the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:SET:STAR`

:CALCulate{1-16}[:SELected]:MARKer:SET:STOP

Description: The command sets the stimulus stop range to the active marker range value on the active trace of the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK:SET:STOP`

:CALCulate{1-16}[:SELected]:MARKer:TSEArch <char>
:CALCulate{1-16}[:SELected]:MARKer:TSEArch?

Description: The command performs a target search on the active trace of the indicated channel. The query outputs the marker search type for target searches on the active trace of the indicated channel.

Cmd Parameters: <char> TARGet | LEFT | RIGHT

Query Parameters: <char> TARG | LEFT | RIGHT

Range: NA

Default Value: TARG

Syntax Example: `:CALC1:MARK:TSEA TARG`

`:CALC1:MARK:TSEA?`

:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TARget <NRf>
:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TARget?

Description: The command sets the marker search target value for target searches on the active trace of the indicated channel. The query outputs the marker search target value for target searches on the active trace of the indicated channel.

See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3>, <NR3>, <NR3> The output parameter depends on the display type.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:CALC1:MARK:TSEA:TAR 1E7`

`:CALC1:MARK:TSEA:TAR?`

:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TRANsition <char>
:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TRANsition?

Description: The command sets the marker search transition value for target searches on the active trace of the indicated channel. The query outputs the marker search transition value for target searches on the active trace of the indicated channel.

Cmd Parameters: <char> POSitive | NEGative | BOTH

Query Parameters: <char> POS | NEG | BOTH

Range: NA

Default Value: BOTH

Syntax Example: `:CALC1:MARK:TSEA:TRAN POS`

`:CALC1:MARK:TSEA:TRAN?`

5-38 :CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem

The :CALCulate{1-16}[:SELected]:MARKer{1-13} subsystem commands provide configuration and control for the indicated marker.

Marker Subsystems

Related marker configuration, control, and reporting commands are described multiple subsystems:

- ":CALCulate{1-16}:DISPlay:MARKer Subsystem" on page 5-17
- ":CALCulate{1-16}:MARKer Subsystem" on page 5-76
- ":CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem" on page 5-107
- ":CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem" on page 5-110
- ":CALCulate{1-16}[:SELected]:MARKer Subsystem" on page 5-165
- ":CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem" on page 5-178
- ":DISPlay Subsystem" on page 5-223

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:MARKer{1-13}:ACTivate

Description: The command makes the indicated marker of the active trace the active marker. No query.

To find if a marker is active, use the query:

```
:CALCulate{1-16}[:SELected]:MARKer:ACTivate?
```

Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

Cmd Parameters: NA

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MARK1:ACT`

:CALCulate{1-16}[:SELected]:MARKer{1-13}:MOVe <char>

Description: The command moves the indicated marker to the indicated location on the active trace on the indicated channel. No query. The available locations are:

- CENTer = Moves the marker to the center of the trace range.
- REFmarker = Moves the marker to the reference marker on the active trace
- STARt = Moves the marker to the start frequency of the active trace
- STOP = Moves the marker to the stop frequency of the active trace

Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

Cmd Parameters: <char> CENTer | REFmarker | STARt | STOP

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK1:MOV CENT`

:CALCulate{1-16}[:SELected]:MARKer{1-13}:SET <char>

Description: The command sets the start, stop, or center range of the display reference level to the indicated marker range/response on the active trace of the indicated channel. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker. No query.

Cmd Parameters: <char> CENTer | REFmarker | STARt | STOP

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK1:SET CENT`

:CALCulate{1-16}[:SELected]:MARKer{1-13}:X <NRf>
:CALCulate{1-16}[:SELected]:MARKer{1-13}:X?

Description: The command enters the frequency, distance, or time of indicated marker on the active trace and turn on. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker. The query outputs the frequency, distance, or time of the indicated marker on the active trace.

Cmd Parameters: <NRf> The input parameter is in Hertz, Meters, or Seconds.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

Range: The range depends on the parameter type setting:

- Frequency = Minimum Instrument Frequency to the Maximum Instrument Frequency
- Time = 1E-9 Seconds to 4E-9 Seconds
- Distance = -29.965E-3 Meters to 1.1988 Meters

Default Value: The default value is always a frequency. The value depends on the installed options:

- If Option 70 – 70 kHz Low End Frequency Extension is installed, the default value = 70 kHz.
- If Option 70 is not installed, the default value = 10 MHz.

Syntax Example: `:CALC1:MARK1:X 2.0E7`

`:CALC1:MARK1:X?`

:CALCulate{1-16}[:SELected]:MARKer{1-13}:Y?

Description: Query only. The query outputs the response value of the indicated marker on the active trace.Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Query Parameters: <NR3> | <NR3>, <NR3> The output parameters depend on the display type. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MARK1:Y?`

:CALCulate{1-16}[:SELected]:MARKer{1-13}[:STATe] <char>
:CALCulate{1-16}[:SELected]:MARKer{1-13}[:STATe]?

Description: The command toggles on/off displaying the indicated marker of the active trace on/off. Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker. Outputs the on/off display status of the indicated marker of the active trace.Markers 1 through 12 are standard measurement markers. Marker 13 is the reference marker.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MARK1 ON`

`:CALC1:MARK1?`

5-39 :CALCulate{1-16}[:SELected]:MATH Subsystem

The :CALCulate{1-16}[:SELected]:MATH subsystem commands provide configuration and control for inter-trace mathematics operations.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:MATH:DATa:FILe:MEMorize

Description: The command stores the data file math to memory of the indicated channel. (This command is only available with Option 45 installed and when Power Sweep [CW] is selected.)

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MATH:DAT:FIL:MEM`

:CALCulate{1-16}[:SELected]:MATH:DISPlay <char>
:CALCulate{1-16}[:SELected]:MATH:DISPlay?

Description: The command selects the trace memory display operation for the active trace of the indicated channel where:

- DATA = Display only the current sweep data
- MEM = Display only the memory data
- DTM = Display both the current sweep data and the memory data at the same time.
- DMM = Combine the sweep data and the memory data mathematically using a addition, subtraction, multiplication, or division and display only the results.
- OFF = Turn the trace display.

The query outputs the trace memory display operation for the active trace of the indicated channel.

Cmd Parameters: <char> DATA | MEM | DTM | DMM | OFF

Query Parameters: <char> DATA | MEM | DTM | DMM | OFF

Range: NA

Default Value: DATA

Syntax Example: `:CALC1:MATH:DISP MEM`

`:CALC1:MATH:DISP?`

:CALCulate{1-16}[:SELected]:MATH:FUNCtion <char>
:CALCulate{1-16}[:SELected]:MATH:FUNCtion?

Description: The command selects the trace memory math operation on the active trace of the indicated channel. The query outputs the trace memory math operation on the active trace of the indicated channel.

Cmd Parameters: <char> ADD | SUBTract | MULTiply | DIVide

Query Parameters: <char> ADD | SUBT | MULT | DIV

Range: NA

Default Value: DIV

Syntax Example: `:CALC1:MATH:FUNC ADD`

`:CALC1:MATH:FUNC?`

:CALCulate{1-16}[:SELected]:MATH:INTErtrace:FUNCtion <char>
:CALCulate{1-16}[:SELected]:MATH:INTErtrace:FUNCtion?

Description: The command selects the inter-trace memory math operation on the active trace of the indicated channel. The result will be displayed on the active trace. The query outputs the inter-trace memory math operation on the active trace of the indicated channel.

Cmd Parameters: <char> ADD | SUBTract | MULTiply | DIVide

Query Parameters: <char> ADD | SUBT | MULT | DIV

Range: NA

Default Value: DIV

Syntax Example: `:CALC1:MATH:INTE:FUNC ADD`

`:CALC1:MATH:INTE:FUNC?`

:CALCulate{1-16}[:SELected]:MATH:INTErtrace:OPERand{1-2}:DEFine <char1>, <char2>
:CALCulate{1-16}[:SELected]:MATH:INTErtrace:OPERand{1-2}:DEFine?

Description: The command sets the trace number and data type for the indicated operand on the active trace of the indicated channel. Note that both parameters must be defined.

The <char1> value sets the trace number from the following selections:

- TR1 = Trace 1
- TR2 = Trace 2
- TR3 = Trace 3
- TR4 = Trace 4
- TR5 = Trace 5
- TR6 = Trace 6
- TR7 = Trace 7
- TR8 = Trace 8
- TR9 = Trace 9
- TR10 = Trace 10
- TR11 = Trace 11
- TR12 = Trace 12
- TR13 = Trace 13
- TR14 = Trace 14
- TR15 = Trace 15
- TR16 = Trace 16

The <char2> value sets the data operand as:

- DATA = Display just the current sweep data
- DMM = Combine the sweep data and the memory data mathematically using a addition, subtraction, multiplication, or division and display only the results.

The query outputs the trace number and data type for the indicated operand on the active trace of the indicated channel.

Cmd Parameters: <char1> TR1 | TR2 | TR3 | TR4 | TR5 | TR6 | TR7 | TR8 | TR9 | TR10 | TR11 | TR12 | TR13 | TR14 | TR15 | TR16

<char2> DATA | DMM

Query Parameters: <char1>, <char2>

Range: NA

Default Value: TR1, DATA

Syntax Example: `:CALC1:MATH:INTE:OPER1:DEF TR1, DATA`

`:CALC1:MATH:INTE:OPER1:DEF?`

:CALCulate{1-16}[:SELected]:MATH:INTErtrace[:STATe] <char>
:CALCulate{1-16}[:SELected]:MATH:INTErtrace[:STATe]?

Description: The command toggles on/off the inter-trace math operation on the active trace of the indicated channel. The query outputs the on/off state of the inter-trace math operation on the active trace of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MATH:INTE ON`

`:CALC1:MATH:INTE?`

:CALCulate{1-16}[:SELected]:MATH:MEMorize

Description: The command stores the active trace data to memory for the channel indicated. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MATH:MEM`

5-40 :CALCulate{1-16}[:SELected]:MDATA Subsystem

The :CALCulate{1-16}[:SELected]:MDATA subsystem provides configuration and control for trace memory data.

Trace Subsystems

Related trace subsystems are:

- “:CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem” on page 5-92
- “:CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem” on page 5-101
- “:CALCulate{1-16}:PARameter{1-16}:SELect Subsystem” on page 5-112
- “:CALCulate{1-16}:POLar Subsystem” on page 5-113
- “:CALCulate{1-16}:PROCessing:ORDer Subsystem” on page 5-115
- “:CALCulate{1-16}[:SELected]:CONVersion Subsystem” on page 5-134
- “:CALCulate{1-16}[:SELected]:DATA Subsystem” on page 5-136
- “:CALCulate{1-16}[:SELected]:FORmat Subsystem” on page 5-147
- “:CALCulate{1-16}[:SELected]:MATH Subsystem” on page 5-181
- “:CALCulate{1-16}[:SELected]:MDATA Subsystem” on page 5-185
- “:CALCulate{1-16}[:SELected]:SMITh Subsystem” on page 5-194
- “:CALCulate{1-16}[:SELected]:SMOothing Subsystem” on page 5-196
- “:CALCulate{1-16}[:SELected]:TDATA Subsystem” on page 5-197
- “:DISPlay Subsystem” on page 5-223

:CALCulate{1-16}[:SELected]:MDATA:FDATa <block>
:CALCulate{1-16}[:SELected]:MDATA:FDATa?

Description: The command inputs formatted trace memory data for the active trace on the indicated channel. The query outputs formatted trace data of the active trace on the indicated channel.

Cmd Parameters: <block> data formatted as XML. See definition of “<block> or <arbitrary block>” on page 2-12. The <block> data must exist.

Query Parameters: <block> data formatted as XML. See definition of “<block> or <arbitrary block>” on page 2-12. The <block> data must exist.

Range: NA

Default Value: NA

Syntax Example:
```
:CALC1:MDATA:FDAT <block>
:CALC1:MDATA:FDAT?
```

:CALCulate{1-16}[:SELected]:MDATA:SDATa <block>
:CALCulate{1-16}[:SELected]:MDATA:SDATa?

Description: The command inputs S-parameter trace memory data for the active trace on the indicated channel. The query outputs S-parameter trace memory data of the active trace on the indicated channel.

Cmd Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:MDATA:SDAT <block>`

`:CALC1:MDATA:SDAT?`

5-41 :CALCulate{1-16}[:SELected]:MSTatistics Subsystem

The `:CALCulate{1-16}[:SELected]:MSTatistics` subsystem provides commands for settings of Time Domain, Eye Diagram marker statistics.

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:AMPLitude[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:AMPLitude[:STATe]?

Description: The command sets the 'eye amplitude' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Amplitude’ selection on the Amplitude Setup menu.

The query outputs the 'eye amplitude' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:AMPL 0`

`:CALC1:MST:EYE:AMPL?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ARELated[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ARELated[:STATe]?

Description: The command sets the display ON/OFF of the amplitude related eye statistics for the given trace of the indicated channel.

The query outputs the display state of the amplitude related eye statistics for the given trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MST:EYE:AREL 1`

`:CALC1:MST:EYE:AREL`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:CPERcentage[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:CPERcentage[:STATe]?

Description: The command sets eye crossing percentage display state to ON/OFF for the active trace of the indicated channel.

The query outputs the eye crossing percentage display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:CPER 1`

`:CALC1:MST:EYE:CPER?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:DCDistortion[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:DCDistortion[:STATe]?

Description: The command sets the 'eye duty cycle distortion' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Duty Cycle Distortion’ selection on the Time Setup menu.

Outputs the 'eye duty cycle distortion' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:DCD 0`

`:CALC1:MST:EYE:DCD?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:FTIMe[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:FTIMe[:STATe]?

Description: The command sets the 'eye fall time' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Fall Time’ selection on the Time Setup menu.

The query outputs the 'eye fall time' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:FTIM 0`

`:CALC1:MST:EYE:FTIM?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:HEIGht[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:HEIGht[:STATe]?

Description: The command sets the 'eye height' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Height’ selection on the Amplitude Setup menu.

The query outputs the 'eye height' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:HEIG 0`

`:CALC1:MST:EYE:HEIG?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MARKer:POSition <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MARKer:POSition?

Description: The command sets the eye marker position on the active trace of the given channel.

The query outputs the eye marker position on the active trace of the given channel.

Cmd Parameters: <char> ULEFt | URIGht | LLEFt | LRIGht | DSOFf

Query Parameters: NA

Query Output: <char> ULEF | URIG | LLEF | LRIG | DSOF

Range: NA

Default Value: ULEF

Syntax Example: `:CALC1:MST:EYE:MARK:POS ULEF`

`:CALC1:MST:EYE:MARK:POS?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MLEVel[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MLEVel[:STATe]?

Description: The command sets the 'eye level mean' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Level Mean’ selection on the Amplitude Setup menu.

The query outputs the 'eye level mean' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:MLEV 0`

`:CALC1:MST:EYE:MLEV?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OFACtor[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OFACtor[:STATe]?

Description: The command sets the 'eye opening factor' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Opening Factor’ selection on the Amplitude Setup menu.

The query outputs the 'eye opening factor' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:OFAC 0`

`:CALC1:MST:EYE:OFAC?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OLEVel[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OLEVel[:STATe]?

Description: The command sets the 'eye level 1' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘1’ Level selection on the Amplitude Setup menu.

The query outputs the 'eye level 1' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:OLEV 0`

`:CALC1:MST:EYE:OLEV?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:PPJitter[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:PPJitter[:STATe]?

Description: The command sets the 'eye PP jitter' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Pk-Pk Jitter’ selection on the Time Setup menu.

The query outputs the 'eye PP jitter' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:PPJ 0`

`:CALC1:MST:EYE:PPJ?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RMSJitter[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RMSJitter[:STATe]?

Description: The command sets the 'eye RMS jitter' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘RMS Jitter’ selection on the Time Setup menu.

The query outputs the 'eye RMS jitter' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:RMSJ 0`

`:CALC1:MST:EYE:RMSJ?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RTIMe[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RTIMe[:STATe]?

Description: The command sets the 'eye rise time' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Rise Time’ selection on the Time Setup menu.

The query outputs the 'eye rise time' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:RTIM 0`

`:CALC1:MST:EYE:RTIM?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:SNRatio[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:SNRatio[:STATe]?

Description: The command sets the 'eye signal to noise' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Signal to Noise’ selection on the Amplitude Setup menu.

The query outputs the 'eye signal to noise' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:SNR 0`

`:CALC1:MST:EYE:SNR?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:TRELated[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:TRELated[:STATe]?

Description: The command sets the display ON/OFF of the time related eye statistics for the given trace of the indicated channel.

The query outputs the display state of the time related eye statistics for the given trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:MST:EYE:TREL 1`

`:CALC1:MST:EYE:TREL?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:WIDth[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:WIDth[:STATe]?

Description: The command sets the 'eye width' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘Width’ selection on the Time Setup menu.

The query outputs the 'eye width' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:WID 0`

`:CALC1:MST:EYE:WID?`

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ZLEVel[:STATe] <char>
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ZLEVel[:STATe]?

Description: The command sets the 'eye level 0' display state to ON/OFF for the active trace of the indicated channel. This corresponds to the ‘0’ Level selection on the Amplitude Setup menu.

The query outputs the 'eye level 0' display state for the active trace of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:CALC1:MST:EYE:ZLEV 0`

`:CALC1:MST:EYE:ZLEV?`

5-42 :CALCulate{1-16}[:SELected]:SMITh Subsystem

The :CALCulate{1-16}[:SELected]:SMITh subsystem commands are used to configure Smith Chart related displays.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:SMITh:IMPedance <NRf>
:CALCulate{1-16}[:SELected]:SMITh:IMPedance?

Description: The command sets the impedance to use for Smith chart related displays on the active trace of the indicated channel. The display type must be Smith chart. The query outputs the impedance to use for Smith chart related displays on the active trace of the indicated channel. The display type must be Smith chart.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPNF

Default Value: 5.000000E+001

Syntax Example: `:CALC1:SMIT:IMP 7.5E1`

`:CALC1:SMIT:IMP?`

:CALCulate{1-16}[:SELected]:SMITh:WRAP <NRf>
:CALCulate{1-16}[:SELected]:SMITh:WRAP?

Description: The command sets the wrap offset to use for a Smith chart display on the active trace of the indicated channel. The display type must be Smith chart. The query outputs the wrap offset to use for a Smith chart display on the active trace of the indicated channel. The display type must be Smith chart.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: -3.6E2 to 3.6E2

Default Value: 0.000000E+000

Syntax Example: `:CALC1:SMIT:WRAP 2.10E1`

`:CALC1:SMIT:WRAP?`

5-43 :CALCulate{1-16}[:SELected]:SMOothing Subsystem

The :CALCulate{1-16}[:SELected]:SMOothing subsystem commands are used to configure and control trace smoothing functions.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:SMOothing:APERture <NRf>
:CALCulate{1-16}[:SELected]:SMOothing:APERture?

Description: The command sets the smoothing aperture for the indicated channel and active trace. The query outputs the smoothing aperture for the indicated channel and active trace.

Cmd Parameters: <NRf> The input parameter is in Percent.

Query Parameters: <NR3> The output parameter is in Percent.

Range: 0 to 100

Default Value: 0.000000E+000

Syntax Example: `:CALC1:SMO:APER 2`

`:CALC1:SMO:APER?`

:CALCulate{1-16}[:SELected]:SMOothing[:STATe] <char>
:CALCulate{1-16}[:SELected]:SMOothing[:STATe]?

Description: The command toggles smoothing on/off for the indicated channel and active trace. The query outputs the smoothing on/off status for the indicated channel and active trace.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:SMO ON`

`:CALC1:SMO?`

5-44 :CALCulate{1-16}[:SELected]:TDATA Subsystem

The :CALCulate{1-16}[:SELected]:TDATA subsystem commands are used to input and report on trace data files.

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- ":CALCulate{1-16}:FORMat Subsystem - SnP Data" on page 5-49
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":FORMat Subsystem" on page 5-242
- ":HCOPy Subsystem" on page 5-245
- ":MMEMory Subsystem" on page 5-251

:CALCulate{1-16}[:SELected]:TDATA:FDATa <block>
:CALCulate{1-16}[:SELected]:TDATA:FDATa?

Description: The command inputs formatted trace data to display on the active trace on the indicated channel. The query outputs formatted trace data of the active trace on the indicated channel.

Cmd Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:TDATA:FDAT <block>`

`:CALC1:TDATA:FDAT?`

:CALCulate{1-16}[:SELected]:TDATA:SDATa <block>
:CALCulate{1-16}[:SELected]:TDATA:SDATa?

Description: The command inputs S-parameter trace data to display on the active trace of the indicated channel. The query outputs S-parameter trace data of the active trace on the indicated channel.

Cmd Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:TDATA:SDAT <block>`

`:CALC1:TDATA:SDAT?`

5-45 :CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem

The :CALCulate{1-16}[:SELected]:TRANSform:TIME subsystem commands are used to configure and control time domain transformation parameters, time domain displays, gate parameters, and window parameters.

Time Domain, Group Delay, and Reference Plane Subsystems

Related time domain, group delay, and reference place subsystems are:

- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:GCOMpression Subsystem" on page 5-149
- ":CALCulate{1-16}[:SELected]:GDELay Subsystem" on page 5-152
- ":CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem" on page 5-199
- ":SENSe{1-16}:CORRection:EXTension Subsystem" on page 5-395

Trace Subsystems

Related trace subsystems are:

- ":CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem" on page 5-92
- ":CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem" on page 5-101
- ":CALCulate{1-16}:PARameter{1-16}:SELect Subsystem" on page 5-112
- ":CALCulate{1-16}:POLar Subsystem" on page 5-113
- ":CALCulate{1-16}:PROCessing:ORDer Subsystem" on page 5-115
- ":CALCulate{1-16}[:SELected]:CONVersion Subsystem" on page 5-134
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:FORmat Subsystem" on page 5-147
- ":CALCulate{1-16}[:SELected]:MATH Subsystem" on page 5-181
- ":CALCulate{1-16}[:SELected]:MDATA Subsystem" on page 5-185
- ":CALCulate{1-16}[:SELected]:SMITh Subsystem" on page 5-194
- ":CALCulate{1-16}[:SELected]:SMOothing Subsystem" on page 5-196
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":DISPlay Subsystem" on page 5-223

:CALCulate{1-16}[:SELected]:TRANsform:TIME:ALIAsfree?

Description: Query only. The query outputs the alias free range of the time domain transform on the active trace of the given channel.

The default for ALIAsfree depends on the VectorStar instrument model number and the installed frequency option, and is calculated based on the formulas below, where:

- AFT = Alias free time
- ST = The frequency step size
- AFT = 0.5 / ST immediately after reset.

For the following models and options, the ST and the AFT are:

- MS4642B without Frequency Extension: ST = (20 GHz minus 10 MHz) divided by 200, AFT = 5.00250125063E–9 seconds;
- MS4642B with Frequency Extension: ST = (20 GHz minus 70 kHz) divided by 200, AFT = 5.00001750006E– seconds;
- MS4644B without Frequency Extension: ST = (40 GHz minus 10 MHz) divided by 200, AFT = 2.50062515629E–9 seconds;
- MS4644B with Frequency Extension: ST = (40 GHz minus 70 kHz) divided by 200, AFT = 2.50000437501E–9 seconds;
- MS4647B without Frequency Extension: ST = (70 GHz minus 10 MHz) divided by 200, AFT = 1.42877553994E–9 seconds;
- MS4647B with Frequency Extension: ST = (70 GHz minus 70 kHz) divided by 200, AFT = 1.42857285714E–9 seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: –1E-9 to 4E-9

Default Value: The default for ALIAsfree depends on the VectorStar instrument model number and the installed frequency option. See the Description above.

Syntax Example: `:CALC1:TRAN:TIME:ALIA?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:CENTer <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:CENTer?

Description: Sets the center time/distance of the range of the time domain transform on the active trace of the given channel. Outputs the center time/distance of the range of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: –999.999 to 999.999 Seconds

–2.99649E11 to 21.99649E11 Meters

Default Value: 1.50000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:CENT 5E2`

`:CALC1:TRAN:TIME:CENT?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm?

Description: Sets the DC term of the time domain transform on the active trace of the given channel.

- Use the AUTO value to allow the VNA instrument to determine the appropriate DC Term value.
- Use the OTHER value to allow for a user-defined DC Term value using the command:

`:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm:OTHer <NRf>`

The query returns the state of the current DC Term type.

Cmd Parameters: <char> AUTO | OTHER

Query Parameters: <char> AUTO | OTHER

Range: NA

Default Value: AUTO

Syntax Example: `:CALC1:TRAN:TIME:DCT AUTO`

`:CALC1:TRAN:TIME:DCT?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm:OTHer <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm:OTHer?

Description: Enters the other value for the DC term of the time domain transform on the active trace of the given channel. Outputs the Other value for the DC Term of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: 0 to 5E3

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:TRAN:TIME:DCT:OTH 5.0E1`

`:CALC1:TRAN:TIME:DCT:OTH?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:DISTance?

Description: Query only. Outputs the list of time domain distance values on the active trace of the given channel.

Query Parameters: <block> or <arbitrary block> See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:TRAN:TIME:DIST?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:EXTrapolate <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:EXTrapolate?

Description: Sets the extrapolation method of the time domain transform on the active trace of the given channel. Outputs the extrapolation method of the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> MAGPHase | PHASE | USER

Query Parameters: <char> MAGPH | PHASE | USER

Range: NA

Default Value: PHASE

Syntax Example: `:CALC1:TRAN:TIME:EXT MAGPH`

`:CALC1:TRAN:TIME:EXT?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:CENTer <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:CENTer?

Description: Sets the center time/distance of the gate of the time domain transform on the active trace of the given channel. Outputs the center time/distance of the gate of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: -999.99 to 999.99 Seconds
-2.99649E11 to 2.99649E11 Meters

Default Value: 1.50000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:GATE:CENT 1.0E-3`

`:CALC1:TRAN:TIME:GATE:CENT?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:DCGamma <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:DCGamma?

Description: Sets the Dolph-Chebyshev gamma value of the time domain transform gate on the active trace of the given channel. Outputs the Dolph-Chebyshev gamma value of the time domain transform gate on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: 0 to 2E2

Default Value: 4.00000000000E+001

Syntax Example: `:CALC1:TRAN:TIME:GATE:DCG 3`

`:CALC1:TRAN:TIME:GATE:DCG?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:KBBeta <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:KBBeta?

Description: Sets the Kaiser-Bessel beta value of the time domain transform gate on the active trace of the given channel. Outputs the Kaiser-Bessel beta value of the time domain transform gate on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: 0 to E308

Default Value: 5.00000000000E-001

Syntax Example: `:CALC1:TRAN:TIME:GATE:KBB 3`

`:CALC1:TRAN:TIME:GATE:KBB?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:NOTCh[:STATe] <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:NOTCh[:STATe]?

Description: Turns anti-gating on/off in the time domain transform on the active trace of the given channel. Outputs the anti-gating on/off status in the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:TRAN:TIME:GATE:NOT ON`

`:CALC1:TRAN:TIME:GATE:NOT?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SHAPe <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SHAPe?

Description: Sets the gate shape of the time domain transform on the active trace of the given channel. The various gate shapes provide different point weighting with resultant changes in the display and its resolution. The available gate shapes are:

- DCHebyshev = Sets a Dolph-Chebyshev window with a trade-off between the side lobe level and resolution. In the DCH, the side lobe level is parameterized in absolute dB where a larger value leads to a wider main lobe width with lower resolution.
- KBEssel = Sets a Kaiser-Bessel window with a trade-off between the side lobe level and resolution. The KBE larger Beta value provides lower side lobes with a wider main lobe width with lower resolution.
- MINimum = This is a rectangular (or null) gate. It will produce the best resolution but the worst side lobe levels.
- NOMinal = The default value which provides about one-half of the resolution with no window but approximately a 30 dB reduction in side lobe levels. This setting advised for mode applications. The NOMinal setting is equivalent to a Hamming gate.
- WIDE = The WIDE gate (Blackman 3 term) will reduce resolution relative to the nominal gate but further reduce side lobe levels.
- MAXimum = The MAXimum gate (Blackman-Harris 4 term) has the poorest resolution of the fixed gates but has the lowest side lobe levels.

Outputs the gate shape of the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> MINimum | NOMinal | WIDE | MAXimum | DCHebyshev | KBEssel

Query Parameters: <char> MIN | NOM | WIDE | MAX | DCH | KBE

Range: NA

Default Value: NOM

Syntax Example: `:CALC1:TRAN:TIME:GATE:SHAP MIN`

`:CALC1:TRAN:TIME:GATE:SHAP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SPAN <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SPAN?

Description: Sets the span time/distance of the gate of the time domain transform on the active trace of the given channel. Outputs the span time/distance of the gate of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: -999.99 to 999.99 Seconds

-2.99649E11 to 2.99649E11 Meters

Default Value: 1.00000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:GATE:SPAN 5.0E-3`

`:CALC1:TRAN:TIME:GATE:SPAN?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STARt <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STARt?

Description: Sets the start time/distance of the gate of the time domain transform on the active trace of the given channel. Outputs the start time/distance of the gate of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: -999.99 to 999.99 Seconds

-2.99649E11 to 2.99649E11 Meters

Default Value: 1.00000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:GATE:STAR 2.0E-3`

`:CALC1:TRAN:TIME:GATE:STAR?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STOP <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STOP?

Description: Outputs the stop time/distance of the gate of the time domain transform on the active trace of the given channel. Sets the stop time/distance of the gate of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: -999.99 to 999.99 Seconds

-2.99649E11 to 2.99649E11 Meters

Default Value: 2.00000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:GATE:STOP 1.0E-2`

`:CALC1:TRAN:TIME:GATE:STOP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE[:STATe] <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE[:STATe]?

Description: Sets the status of the gate of the time domain transform on the active trace of the given channel where:

- ON = The selected gate is turned on and the display adjusted appropriately.
- OFF = The gate is turned off, and the display returns to its prior state.
- DISPLAY = The upper and lower bounds of the gate are shown along with the trace display in its normal state. Use this setting to help position the gate accurately on the trace.

Cmd Parameters: <char> ON | OFF | DISPlay

Query Parameters: <char> ON | OFF | DISP

Range: NA

Default Value: OFF

Syntax Example: `:CALC1:TRAN:TIME:GATE ON`

`:CALC1:TRAN:TIME:GATE?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:IMPUlsewidth?

Description: Query only. Outputs the impulse width of the time domain transform on the active trace of the given channel.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: NA

Default Value: 0.00000000000E+000

Syntax Example: `:CALC1:TRAN:TIME:IMPU?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:PHASor:IMPulse[:STATe] <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:PHASor:IMPulse[:STATe]?

Description: Turns the Phasor Impulse mode on/off state in the time domain transform on the active trace of the given channel. The query returns the Phasor Impulse mode on/off state in the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:TRAN:TIME:PHAS:IMP ON`

`:CALC1:TRAN:TIME:PHAS:IMP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:RESPonse <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:RESPonse?

Description: Sets the response of the time domain transform on the active trace of the given channel. Outputs the response of the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> IMPulse | STEP

Query Parameters: <char> IMP | STEP

Range: NA

Default Value: IMP

Syntax Example: `:CALC1:TRAN:TIME:RESP STEP`

`:CALC1:TRAN:TIME:RESP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:SPAN <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:SPAN?

Description: Sets the span time/distance of the range of the time domain transform on the active trace of the given channel. Outputs the span time/distance of the range of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: -999.99 to 999.99 Seconds
-2.99649E11 to 2.99649E11 Meters

Default Value: 5.00000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:SPAN 5.0E-3`

`:CALC1:TRAN:TIME:SPAN?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:STARt <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:STARt?

Description: Sets the start time/distance of the range of the time domain transform on the active trace of the given channel. Outputs the start time/distance of the range of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: -999.99 to 999.99 Seconds
-2.99649E11 to 2.99649E11 Meters

Default Value: -1.00000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:STAR 2.0E-3`

`:CALC1:TRAN:TIME:STAR?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:STOP <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:STOP?

Description: Sets the stop time/distance of the range of the time domain transform on the active trace of the given channel. Outputs the stop time/distance of the range of the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is in Seconds or Meters.

Query Parameters: <NR3> The output parameter is in Seconds or Meters.

Range: -999.99 to 999.99 Seconds
-2.99649E11 to 2.99649E11 Meters

Default Value: 4.00000000000E-009

Syntax Example: `:CALC1:TRAN:TIME:STOP 1.0E-2`

`:CALC1:TRAN:TIME:STOP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:TIME?

Description: Query only. Outputs the list of time domain time values on the active trace of the given channel.

Query Parameters: See definition of “<block> or <arbitrary block>” on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:CALC1:TRAN:TIME:TIME?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:TRIP <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:TRIP?

Description: Sets the trip length of the time domain transform on the active trace of the given channel. Outputs the trip length of the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> ONEway | ROUNDtrip | AUTO

Query Parameters: <char> ONE | ROUND | AUTO

Range: NA

Default Value: AUTO

Syntax Example: `:CALC1:TRAN:TIME:TRIP ONE`

`:CALC1:TRAN:TIME:TRIP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:TYPe <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:TYPe?

Description: Sets the time domain transform type on the active trace of the given channel. Outputs the time domain transform type on the active trace of the given channel.

Cmd Parameters: <char> FREQuency | FREQGATE | LOWpass | BANDpass | EYEdiagram

Query Output: <char> FREQ | FREQGATE | LOW | BAND | EYE

Range: NA

Default Value: FREQ

Syntax Example: `:CALC1:TRAN:TIME:TYP EYE`

`:CALC1:TRAN:TIME:TYP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:UNIt <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:UNIt?

Description: Sets the unit used in the time domain transform on the active trace of the given channel. Outputs the unit used in time domain transform on the active trace of the given channel.

Cmd Parameters: <char> TIME | DISTance

Query Output: <char> TIME | DIST

Range: NA

Default Value: TIME

Syntax Example: `:CALC1:TRAN:TIME:UNI DIST`

`:CALC1:TRAN:TIME:UNI?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:DCGamma <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:DCGamma?

Description: Sets the Dolph-Chebyshev gamma value of the time domain transform window on the active trace of the given channel. Outputs the Dolph-Chebyshev gamma value of the time domain transform window on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: 0 to 2E2

Default Value: 4.00000000000E+001

Syntax Example: `:CALC1:TRAN:TIME:WIND:DCG 3`

`:CALC1:TRAN:TIME:WIND:DCG?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:KBBeta <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:KBBeta?

Description: Sets the Kaiser-Bessel beta value of the time domain transform window on the active trace of the given channel. Outputs the Kaiser-Bessel beta value of the time domain transform window on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: 0 to E308

Default Value: 5.00000000000E-001

Syntax Example: `:CALC1:TRAN:TIME:WIND:KBB 3`

`:CALC1:TRAN:TIME:WIND:KBB?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:SHAPe <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:SHAPe?

Description: Sets the time window shape of the time domain transform on the active trace of the given channel. The various time window shapes provide different point weighting with resultant changes in the display and its resolution. The available time window shapes are:

- RECTangular
- NOMinal
- LOWsidelobe
- MINsidelobe
- DCHebyshev
- KBEssel

Outputs the window shape of the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> RECTangular | NOMinal | LOWsidelobe | MINsidelobe | DCHebyshev | KBEssel

Query Output: <char> RECT | NOM | LOW | MIN | DCH | KBE

Range: NA

Default Value: NOM

Syntax Example: `:CALC1:TRAN:TIME:WIND:SHAP RECT`

`:CALC1:TRAN:TIME:WIND:SHAP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM:IMPulses <NRf>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM:IMPulses?

Description: Sets the number of zoom impulses in the time domain transform on the active trace of the given channel. Outputs the number of zoom impulses in the time domain transform on the active trace of the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MNPI.

Default Value: 0

Syntax Example: `:CALC1:TRAN:TIME:ZOOM:IMP 5`

`:CALC1:TRAN:TIME:ZOOM:IMP?`

:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM[:STATe] <char>
:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM[:STATe]?

Description: Turns time zoom mode on/off in the time domain transform on the active trace of the given channel. Outputs the time zoom mode on/off status in the time domain transform on the active trace of the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:TRAN:TIME:ZOOM ON`

`:CALC1:TRAN:TIME:ZOOM?`

5-46 :CALCulate{1-16}:SNPSetup Subsystem

The :CALCulate{1-16}:SNPSetup Subsystem provides an option to save gated data into SnP files.

:CALCulate{1-16}:SNPSetup:GATed:DATa[:STATe] <char>
:CALCulate{1-16}:SNPSetup:GATed:DATa[:STATe]

Description: Turns on/off the option for saving gated data into .sNp files on the given channel. The The query outputs the On/Off state of the option for saving gated data into .sNp files on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Range: NA

Default Value: 0

Syntax Example: :CALC1:SNPS:GAT:DAT ON

:CALC1:SNPS:GAT:DAT?

5-47 :CALCulate{1-16}:TMS Subsystem

The :CALCulate{1-16}:TMS subsystem commands control the true mode stimulus configuration within the DifferentialView™ application.

:CALCulate{1-16}:TMS:MODe <char>
:CALCulate{1-16}:TMS:MODe?

Description: Configures the stimulus mode on the indicated channel. Returns the stimulus mode on the indicated channel.

Cmd Parameters: <char> TMSO | TMSD | TMSR | TMSS
TMSO = TMS Off (Single Ended)
TMSD = TMS Drive
TMSR = TMS Reverse
TMSS = Source (Ref Plane)

Query Parameters: <char> TMSO | TMSD | TMSR | TMSS

Range: NA

Default Value: TMSO

Syntax Example: `CALC1:TMS:MOD TMSS`

`CALC1:TMS:MOD?`

:CALCulate{1-16}:TMS:PHASe:SWEep[:STATe] <char>
:CALCulate{1-16}:TMS:PHASe:SWEep[:STATe]?

Description: Turns on/off the state of phase sweep on the indicated channel. Outputs the on/off state of the phase sweep on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CALC1:TMS:PHAS:SWE ON`

`:CALC1:TMS:PHAS:SWE?`

:CALCulate{1-16}:TMS:PHASe:SWEep:POINts <NRf>
:CALCulate{1-16}:TMS:PHASe:SWEep:POINts?

Description: Sets the phase sweep number of points on the indicated channel. Returns the phase sweep number of points on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unit less number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 2 to instrument number of points setting (25000 or 100000).

Default Value: 2

Syntax Example: `:CALC1:TMS:PHAS:SWE:POIN 201`

`:CALC1:TMS:PHAS:SWE:POIN?`

:CALCulate{1-16}:TMS:PHASe:SWEep:STARt <NRf>
:CALCulate{1-16}:TMS:PHASe:SWEep:STAR?

Description: Sets the phase sweep start phase on the indicated channel. Returns the phase sweep start phase on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in degrees.

Query Parameters: <NR3> The output parameter is in degrees.

Range: –3600 to +3600

Default Value: 0

Syntax Example: `:CALC1:TMS:PHAS:SWE:STAR 3.0E1`

`:CALC1:TMS:PHAS:SWE:STAR?`

:CALCulate{1-16}:TMS:PHASe:SWEep:STOP <NRf>
:CALCulate{1-16}:TMS:PHASe:SWEep:STOP?

Description: Sets the phase sweep stop phase on the indicated channel. Returns the phase sweep stop phase on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in degrees.

Query Parameters: <NR3> The output parameter is in degrees.

Range: –3600 to +3600

Default Value: 180.00

Syntax Example: `:CALC1:TMS:PHAS:SWE:STOP 1.8E2`

`:CALC1:TMS:PHAS:SWE:STOP?`

:CALCulate{1-16}:TMS:POWer:OFFSet <NRf>
:CALCulate{1-16}:TMS:POWer:OFFSet?

Description: Sets the TMS power offset on the indicated channel. Returns the TMS power offset on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: –50 dB to +50 dB

Default Value: 0

Syntax Example: `:CALC1:TMS:POW:OFFS 10`

`:CALC1:TMS:POW:OFFS?`

:CALCulate{1-16}:TMS:REFerence:PORT <char>
:CALCulate{1-16}:TMS:REFerence:PORT?

Description: Selects the reference port of the indicated channel. Outputs the reference port of the indicated channel.

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4

Query Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4

Range: NA

Default Value: PORT1

Syntax Example: `:CALC1:TMS:REF:PORT PORT2`

`:CALC1:TMS:REF:PORT?`

5-48 :CONTrol:AUXio Subsystem

The :CONTrol:AUXio subsystem commands controls the rear panel limit failure configuration and display.

Rear Panel Subsystems

A related rear panel subsystem is:

- ":CONTrol{1-16}:AOUT Subsystem" on page 5-216

:CONTrol:AUXio:LIMit:POLarity <char>
:CONTrol:AUXio:LIMit:POLarity?

Description: Sets the Positive/Negative Polarity of the Limit Fail indication on the Rear Panel Auxiliary IO connector. Outputs the Positive/Negative Polarity of the Limit Fail indication on the Rear Panel Auxiliary IO connector.

Cmd Parameters: <char> POSitive | NEGative

Query Parameters: <char> POS | NEG

Range: NA

Default Value: POS

Syntax Example: `:CONT:AUX:LIM:POL POS`

`:CONT:AUX:LIM:POL?`

:CONTrol:AUXio:LIMit[:STATe] <char>
:CONTrol:AUXio:LIMit[:STATe]?

Description: Turns on/off indicating a limit failure on the Rear Panel Auxiliary I/O connector. Outputs the on/off status of indication of a limit failure on the Rear Panel Auxiliary I/O connector.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CONT:AUX:LIM ON`

`:CONT:AUX:LIM?`

5-49 :CONTrol{1-16}:AOUT Subsystem

The :CONTrol{1-16}:AOUT subsystem commands allows the user to specify the rear panel output mode, and then set appropriate voltage levels, pulse types, and ports for that mode. When all parameters are entered, a single command allows the rear panel output to be toggled on and off. The settings are saved and can be recalled with the four setup file types:

- Active Channel Setup and Cal (*.chx)
- Active Channel Setup (*.stx)
- All Channel Setup and Cal (*.cha)
- All Channel Setup (*.sta)

Rear Panel Subsystems

A related rear panel subsystem is:

- ":CONTrol:AUXio Subsystem" on page 5-215

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- ":CALCulate{1-16}:FORMat Subsystem - SnP Data" on page 5-49
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":FORMat Subsystem" on page 5-242
- ":HCOPy Subsystem" on page 5-245
- ":MMEMory Subsystem" on page 5-251

Setting the Rear Panel Output Mode as Horizontal

For example, to set the Rear Panel Output Mode at Horizontal, follow the general sequence below. In this example, the control is for Channel 4 with a Start voltage of -1.5 V and a Stop voltage of 2.8 V.

1. Turn the Rear Panel Output off:

   `:CONTrol4:AOUT:STATe OFF`

2. Set the Output Mode to HORizontal:

   `:CONTrol4:AOUT:MODE HORizontal`

3. Set the Horizontal Output Mode start (STARt) voltage to -1.5 V:

   `:CONTrol4:AOUT:VOLTage:STARt -1.500`

4. Set the Horizontal Output Mode stop (STOP) voltage to +2.8 V:

   `:CONTrol4:AOUT:VOLTage:STOP 2.800`

5. Check voltage settings:

   `:CONTrol4:AOUT:VOLTage:STARt?`

   `-1.500`

   `:CONTrol4:AOUT:VOLTage:STOP?`

   `2.800`

6. Turn the Rear Panel Output on:

   `:CONTrol4:AOUT:STATe ON`

7. Check that the Rear Panel is on:

   `:CONTrol4:AOUT:STATe?`

   `1`

Setting the Rear Panel Output Mode as Driven Port

To set the Rear Panel Output Mode as Driven Port, follow the general command sequence below. In this example, the control is for Channel 2, with Port 1 at 3 V and Port 2 at 5 V.

1. Turn the Rear Panel Output off:

   `:CONTrol4:AOUT:STATe OFF`

2. Set the Output Mode to DRIVen

   `:CONTrol4:AOUT:MODE DRIVen`

3. Set the Driven Output Voltage Level for Port 1 to 3 V:

   `:CONTrol4:AOUT1:DRIVen:LEV 3.000`

4. Set the Driven Output Voltage Level for Port 2 to 5 V:

   `:CONTrol4:AOUT2:DRIVen:LEV 5.000`

5. Check voltage settings:

   `:CONTrol4:AOUT1:DRIVen:LEV?`

   `3.000`

   `:CONTrol4:AOUT2:DRIVen:LEV?`

   `5.000`

6. Turn the Rear Panel Output on:

   `:CONTrol{1-16}:AOUT:STATe ON`

7. Check that the Rear Panel is on:

   `:CONTrol{1-16}:AOUT:STATe?`

   `1`

Setting the Rear Panel Output Mode as TTL

To set the Rear Panel Output Mode as TTL, follow the general command sequence below. In this example, the control is for Channel 5, with Port 1 at High, and Port 2 at Low Pulse set to 5.000 ms.

1. Turn the Rear Panel Output off:

   `:CONTrol5:AOUT:STATe OFF`

2. Set the Output Mode to TTL:

   `:CONTrol5:AOUT:MODE TTL`

3. Set the TTL Output for Port 1 as High (HIGH):

   `:CONTrol5:AOUT1}:TTL:TYPe HIGH`

4. Set the TTL Output for Port 2 as Low Pulse (LPULSE):

   `:CONTrol5:AOUT1}:TTL:TYPe LPULSE`

5. Using units of seconds, set the Port 2 TTL Pulse Width to 0.001 seconds:

   `:CONTrol5:AOUT:PULSe:WIDth 1.0E-3`

6. Check TTL and TTL Pulse Width settings:

```
:CONTrol5:AOUT:MODE?
TTL
:CONTrol5:AOUT1:TTL:TYPe?
HIGH
:CONTrol5:AOUT2:TTL:TYPe?
LPULSE
:CONTrol5:AOUT:PULSe:WIDth?
1.0E-3
```

7. Turn the Rear Panel Output on:

```
:CONTrol5:AOUT:STATe ON
```

8. Check that the Rear Panel is on:

```
:CONTrol5:AOUT:STATe?
1
```

:CONTrol{1-16}:AOUT:MODE <char>
:CONTrol{1-16}:AOUT:MODE?

Description: For the indicated channel, the command sets the Rear Panel Output mode to horizontal, driven, TTL, or vertical where:

- HORizontal = Defines the rear panel output voltage to be proportional to the horizontal position on the screen. If HOR is set, the Start and Stop voltages must by set, where the Start Voltage is the voltage for the first sweep point, and the Stop Voltage is the voltage for the last sweep point. The Start Voltage can be higher than, equal to, or less than the Stop Voltage. The voltage range for each is -10 VDC to +10 VDC. The Start Voltage is set by the :CONTrol{1-16}:AOUT:VOLTage:STARt command while the Stop Voltage is set by the :CONTrol{1-16}:AOUT:VOLTage:STOP command (both described below).
- DRIVen = Defines the rear panel port analog output voltage as a function of the currently driven port. For example, if Port 1 is driving, the selected voltage is that set for Port 1. If Port 2 is driving, the selected voltage is that set for Port 2. If DRIV is set, the Output Level Voltage for Port 1 or Port 2 must be set. The command :CONTrol{1-16}:AOUT{1-2}:DRIVen:LEV sets the driven port voltage level for the indicated channel and port with a voltage range of -10 VDC to +10 VDC.
- TTL = Defines the TTL power for each port as steady state high or low, or as pulse high or low. If pulse is selected, the duration of the pulse is also defined. The command :CONTrol{1-16}:AOUT{1-2}:TTL:TYPe defines the TTL type used as HIGH (+5 VDC), LOW (0 VDC), HIGH PULSE (+5 VDC), or LOW PULSE (0 VDC). If High Pulse or Low Pulse is selected, the Pulse Width in seconds must be specified using the command :CONTrol{1-16}:AOUT:PULSe:WIDth. Note that the TTL type is defined separately for Port 1 and for Port 2.
- VERTical = Defines the rear panel output voltage to represent the vertical magitude of a trace in a scaled fashion. If VERT is set for a rectangular display, the Minimum and Maximum voltages must by set, where the Min Voltage corresponds to the bottom graticule and the Max Voltage corresponds to the top graticule. If set for a circular display, the Min Voltage corresponds to the center graticule and the Max Voltage corresponds to the outer graticule. The Min Voltage can be higher than, equal to, or less than the Max Voltage. The voltage range for each is -10 VDC to +10 VDC. The Min Voltage is set by the :CONTrol{1-16}:AOUT:VOLTage:VMIN command, while the Max Voltage is set by the :CONTrol{1-16}:AOUT:VOLTage:VMAX command (both described below).

The query outputs the currently set Rear Panel Output mode on the indicated channel.

Cmd Parameters: <char> HORizontal | DRIVen | TTL | VERTical

Query Parameters: <char> HOR | DRIV | TTL | VERT

Range: NA

Default Value: HOR

Syntax Example: `:CONT1:AOUT:MODE HOR`

`:CONT1:AOUT:MODE?`

:CONTrol{1-16}:AOUT:PULSe:WIDth <NRf>
:CONTrol{1-16}:AOUT:PULSe:WIDth?

Description: If the Rear Panel Output mode is set to TTL, and a High Plus and/or a Low Pulse is configured, use this command to set the pulse width in seconds for the indicated channel. The query outputs the Rear Panel Output pulse width in units of seconds for the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 0 to 10 seconds

Default Value: 0.000000E+000

Syntax Example: `:CONT1:AOUT:PULS:WID 1.0E-3`

`:CONT1:AOUT:PULS:WID?`

:CONTrol{1-16}:AOUT:VERTical:TRACe <char>
:CONTrol{1-16}:AOUT:VERTical:TRACe?

Description: Sets the rear panel analog output vertical mode trace number (tr1 - tr16). The query outputs the rear panel analog vertical output mode trace number in use.

Cmd Parameters: <char> TR1 | TR2 | TR3 | TR4 | TR5 | TR6 | TR7 | TR8 | TR9 | TR10 | TR11 | TR12 | TR13 | TR14 | TR15 | TR16

Query Parameters: <char> TR1 | TR2 | TR3 | TR4 | TR5 | TR6 | TR7 | TR8 | TR9 | TR10 | TR11 | TR12 | TR13 | TR14 | TR15 | TR16

Range: NA

Default Value: TR1

Syntax Example: `:CONT1:AOUT:VERT:TRAC TR5`

`:CONT1:AOUT:VERT:TRAC?`

:CONTrol{1-16}:AOUT:VERTical:TRACe:ACTive[:STATe] <char>
:CONTrol{1-16}:AOUT:VERTical:TRACe:ACTive[:STATe]?

Description: Sets the rear panel analog output vertical mode user active trace check box to ON/OFF. The query outputs the rear panel analog vertical output mode user active trace checkbox status.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: N/A

Default Value: 1

Syntax Example: `:CONT1:AOUT:VERT:TRAC:ACT ON`

`:CONT1:AOUT:VERT:TRAC:ACT?`

:CONTrol{1-16}:AOUT:VOLTage:STARt <NRf>
:CONTrol{1-16}:AOUT:VOLTage:STARt?

Description: If the Rear Panel Output mode is set to horizontal, this command sets the start voltage for the indicated channel. The Start Voltage is the voltage for the first sweep point in the range of -10 VDC to +10 VDC. The Start Voltage can be lower than, equal to, or greater than the Stop Voltage. The query outputs the Rear Panel Output horizontal start voltage setting for the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Volts.

Query Parameters: <NR3> The output parameter is in Volts.

Range: -10 to +10 Volts

Default Value: 0.00000000000E+000

Syntax Example: `:CONT1:AOUT:VOLT:STAR 0.5`

`:CONT1:AOUT:VOLT:STAR?`

:CONTrol{1-16}:AOUT:VOLTage:STOP <NRf>
:CONTrol{1-16}:AOUT:VOLTage:STOP?

Description: If the Rear Panel Output mode is set to horizontal, this command sets the stop voltage for the indicated channel. The Stop Voltage is the voltage for the last sweep point in the range off -10 VDC to +10 VDC. The Stop Voltage can be lower than, equal to, or greater than the Start Voltage. The query outputs the Rear Panel Output horizontal stop voltage for the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Volts.

Query Parameters: <NR3> The output parameter is in Volts.

Range: -10 to +10 Volts

Default Value: 1.000000E+000

Syntax Example: `:CONT1:AOUT:VOLT:STOP 0.7`

`:CONT1:AOUT:VOLT:STOP?`

:CONTrol{1-16}:AOUT:VOLTage:VMAX <NRf>
:CONTrol{1-16}:AOUT:VOLTage:VMAX?

Description: If the Rear Panel Output mode is set to vertical, this command sets the rear panel analog output vertical mode maximum voltage for the indicated channel. The Max Voltage corresponds to the top display graticule. The query outputs the rear panel analog vertical output mode maximum voltage of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Volts.

Query Parameters: <NR3> The output parameter is in Volts.

Range: –10 to +10 Volts

Default Value: 1.000000E+000

Syntax Example: `:CONT1:AOUT:VOLT:VMAX 5.0`

`:CONT1:AOUT:VOLT:VMAX?`

:CONTrol{1-16}:AOUT:VOLTage:VMIN <NRf>
:CONTrol{1-16}:AOUT:VOLTage:VMIN?

Description: If the Rear Panel Output mode is set to vertical, this command sets the rear panel analog output vertical mode minimum voltage for the indicated channel. The Min Voltage corresponds to the bottom display graticule. The query outputs the rear panel analog vertical output mode minimum start voltage of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Volts.

Query Parameters: <NR3> The output parameter is in Volts.

Range: –10 to +10 Volts

Default Value: 0.000000E+000

Syntax Example: `:CONT1:AOUT:VOLT:VMIN -5.0`

`:CONT1:AOUT:VOLT:VMIN?`

:CONTrol{1-16}:AOUT[:STATe] <char>
:CONTrol{1-16}:AOUT[:STATe]?

Description: Turns rear panel analog output on/off for the indicated channel. Outputs the rear panel analog output on/off status of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:CONT1:AOUT ON`

`:CONT1:AOUT?`

:CONTrol{1-16}:AOUT{1-2}:DRIVen:LEV <NRf>
:CONTrol{1-16}:AOUT{1-2}:DRIVen:LEV?

Description: If the Rear Panel Output mode is set to driven, this command sets the voltage level for the indicated port on the indicated channel. The query outputs the Rear Panel Output driven port voltage level for the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Volts.

Query Parameters: <NR3> The output parameter is in Volts.

Range: -10 to +10 Volts

Default Value: 0.000000E+000

Syntax Example: `:CONT1:AOUT1:DRIV:LEV 0.5`

`:CONT1:AOUT1:DRIV:LEV?`

:CONTrol{1-16}:AOUT{1-2}:TTL:TYPe <char>
:CONTrol{1-16}:AOUT{1-2}:TTL:TYPe?

Description: The command sets the Rear Panel Output mode as TTL type for the indicated port on the indicated channel where:

- HIGH = A TTL high value of +5 VDC
- LOW = A TTL low value of 0 VDC
- HIGH PULSE = A TTL high value of +5 VDC for a pulse width duration in seconds as set by the command `:CONTrol{1-16}:AOUT:PULSe:WIDth`.
- LOW PULSE = A TTL low value of 0 VDC for a pulse width duration in seconds set by the command `:CONTrol{1-16}:AOUT:PULSe:WIDth`.

Note that the TTL type is defined separately for Port 1 and for Port 2. Outputs the rear panel analog output TTL output type of the indicated channel.

Cmd Parameters: <char> LOW | HIGH | LPULSE | HPULSE

Query Parameters: <char> LOW | HIGH | LPULSE | HPULSE

Range: NA

Default Value: HIGH

Syntax Example: `:CONT1:AOUT1:TTL:TYP LOW`

`:CONT1:AOUT1:TTL:TYP?`

5-50 :DISPlay Subsystem

The :DISplay subsystem commands are used to control the VNA graphic display information on a per-instrument, per-channel, and per-trace basis.

Trace Subsystems

Related trace subsystems are:

- “:CALCulate{1-16}:PARameter and :PARameter{1-16} Subsystem” on page 5-92
- “:CALCulate{1-16}:PARameter{1-16}:FSIMulator Subsystem” on page 5-101
- “:CALCulate{1-16}:PARameter{1-16}:SELect Subsystem” on page 5-112
- “:CALCulate{1-16}:POLar Subsystem” on page 5-113
- “:CALCulate{1-16}:PROCessing:ORDer Subsystem” on page 5-115
- “:CALCulate{1-16}[:SELected]:CONVersion Subsystem” on page 5-134
- “:CALCulate{1-16}[:SELected]:DATA Subsystem” on page 5-136
- “:CALCulate{1-16}[:SELected]:FORmat Subsystem” on page 5-147
- “:CALCulate{1-16}[:SELected]:MATH Subsystem” on page 5-181
- “:CALCulate{1-16}[:SELected]:MDATA Subsystem” on page 5-185
- “:CALCulate{1-16}[:SELected]:SMITh Subsystem” on page 5-194
- “:CALCulate{1-16}[:SELected]:SMOothing Subsystem” on page 5-196
- “:CALCulate{1-16}[:SELected]:TDATA Subsystem” on page 5-197
- “:DISPlay Subsystem” on page 5-223

Marker Subsystems

Related marker configuration, control, and reporting subsystems are:

- “:CALCulate{1-16}:DISPlay:MARKer Subsystem” on page 5-17
- “:CALCulate{1-16}:MARKer Subsystem” on page 5-76
- “:CALCulate{1-16}:PARameter{1-16}:MARKer Subsystem” on page 5-107
- “:CALCulate{1-16}:PARameter{1-16}:MSTatistics Subsystem” on page 5-110
- “:CALCulate{1-16}[:SELected]:MARKer Subsystem” on page 5-165
- “:CALCulate{1-16}[:SELected]:MARKer{1-13} Subsystem” on page 5-178
- “:DISPlay Subsystem” on page 5-223

Limit Line Subsystems

Related limit line subsystems are:

- “:CALCulate{1-16}[:SELected]:LIMit Subsystem” on page 5-153
- “:DISPlay Subsystem” on page 5-223

:DISPlay:ACTive:CHANnel:SWEep[:STATe] <char>
:DISPlay:ACTive:CHANnel:SWEep[:STATe]?

Description: The command turns On/Off the option to sweep only the active channel.

The Query outputs the On/Off state of the option to sweep only the active channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Range: NA

Default Value: 0

Syntax Example: `:DISP:ACT:CHAN:SWE:STAT ON`

`:DISP:ACT:CHAN:SWE:STAT?`

:DISPlay:COLor:INVert:BACK <NRf>, <NRf>, <NRf>
:DISPlay:COLor:INVert:BACK?

Description: Sets the inverted color of the background on the active channel. The invert color should be on before using this command.

Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the inverted RGB color of the background on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 255,255,255

Syntax Example: `:DISP:COL:INV:BACK 0, 0, 0`

`:DISP:COL:INV:BACK?`

:DISPlay:COLor:INVert:GRATicule:MAIN <NRf>, <NRf>, <NRf>
:DISPlay:COLor:INVert:GRATicule:MAIN?

Description: Sets the inverted color of the main graticule on the active channel.

Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the inverted RGB color of the main graticule on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 0,0,0

Syntax Example: `:DISP:COL:INV:GRAT:MAIN 255, 0, 0`

`:DISP:COL:INV:GRAT:MAIN?`

:DISPlay:COLor:INVert:GRATicule:SUB <NRf>, <NRf>, <NRf>
:DISPlay:COLor:INVert:GRATicule:SUB?

Description: Sets the inverted color of the subgraticule on the active channel.

Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the inverted RGB color of the subgraticule on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 100, 100, 100

Syntax Example: `:DISP:COL:INV:GRAT:SUB 255, 255, 0`

`:DISP:COL:INV:GRAT:SUB?`

:DISPlay:COLor:INVert:LIMit <NRf>, <NRf>, <NRf>
:DISPlay:COLor:INVert:LIMit?

Description: Sets the inverted color of the limit lines on the active channel.

Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the inverted RGB color of the limit lines on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 255,0,0

Syntax Example: `:DISP:COL:INV:LIM 0, 255, 255`

`:DISP:COL:INV:LIM?`

:DISPlay:COLor:INVert:TRACe{1-16}:DATA <NRf>, <NRf>, <NRf>
:DISPlay:COLor:INVert:TRACe{1-16}:DATA?

Description: Sets the inverted color of the indicated data trace on the active channel.

Use the command below to reset the display to the factory default:

- :DISPlay:COLor:RESet

Outputs inverted RGB color of the indicated data trace on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 255, 255, 0

Syntax Example: `:DISP:COL:INV:TRAC1:DATA 255, 0, 255`

`:DISP:COL:INV:TRAC1:DATA?`

:DISPlay:COLor:INVert:TRACe{1-16}:MEMory <NRf>, <NRf>, <NRf>
:DISPlay:COLor:INVert:TRACe{1-16}:MEMory?

Description: Sets the inverted color of the indicated memory trace on the active channel. Memory trace must be on. Use the command below to reset the display to the factory default:

- :DISPlay:COLor:RESet

Outputs inverted RGB color of the indicated memory trace on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 175, 143, 47

Syntax Example: `:DISP:COL:INV:TRAC1:MEM 0, 0, 255`

`:DISP:COL:INV:TRAC1:MEM?`

:DISPlay:COLor:INVert[:STATe] <char>
:DISPlay:COLor:INVert[:STATe]?

Description: Sets screen object colors to their inverted/normal color state. Use the command below to reset the display to the factory default:

- :DISPlay:COLor:RESet

Outputs the inverted/normal color state of screen objects.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:DISP:COL:INV ON`

`:DISP:COL:INV?`

:DISPlay:COLor:NORMal:BACK <NRf>, <NRf>, <NRf>
:DISPlay:COLor:NORMal:BACK?

Description: Sets the normal color of the background on the active channel. Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the normal RGB color of the background on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 0, 0, 0

Syntax Example: `:DISP:COL:NORM:BACK 255, 0, 255`

`:DISP:COL:NORM:BACK?`

:DISPlay:COLor:NORMal:GRATicule:MAIN <NRf>, <NRf>, <NRf>
:DISPlay:COLor:NORMal:GRATicule:MAIN?

Description: Sets the normal color of the main graticule on the active channel. Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the normal RGB color of the main graticule on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 255, 255, 255

Syntax Example: `:DISP:COL:NORM:GRAT:MAIN 255, 255, 255`

`:DISP:COL:NORM:GRAT:MAIN?`

:DISPlay:COLor:NORMal:GRATicule:SUB <NRf>, <NRf>, <NRf>
:DISPlay:COLor:NORMal:GRATicule:SUB?

Description: Sets the normal color of the sub graticule on the active channel. Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the normal RGB color of the sub graticule on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 100, 100, 100

Syntax Example: `:DISP:COL:NORM:GRAT:SUB 45, 45, 45`

`:DISP:COL:NORM:GRAT:SUB?`

:DISPlay:COLor:NORMal:LIMit <NRf>, <NRf>, <NRf>
:DISPlay:COLor:NORMal:LIMit?

Description: Sets the normal color of the limit lines on the active channel. Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs the normal RGB color of the limit lines on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 255, 0, 0

Syntax Example: `:DISP:COL:NORM:LIM 100, 0, 0`

`:DISP:COL:NORM:LIM?`

:DISPlay:COLor:NORMal:TRACe{1-16}:DATA <NRf>, <NRf>, <NRf>
:DISPlay:COLor:NORMal:TRACe{1-16}:DATA?

Description: Sets the normal color of the indicated data trace on the active channel. Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs normal RGB color of the indicated data trace on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 255, 255, 0

Syntax Example: `:DISP:COL:NORM:TRAC1:DATA 100, 100, 0`

`:DISP:COL:NORM:TRAC1:DATA?`

:DISPlay:COLor:NORMal:TRACe{1-16}:MEMory <NRf>, <NRf>, <NRf>
:DISPlay:COLor:NORMal:TRACe{1-16}:MEMory?

Description: Sets the normal color of the indicated memory trace on the active channel. Use the command below to reset the display to the factory default:

- `:DISPlay:COLor:RESet`

Outputs normal RGB color of the indicated memory trace on the active channel.

Cmd Parameters: <NRf> The input parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Query Parameters: <NR1>, <NR1>, <NR1> The output parameters are integers between 0 and 255 representing the Red, Green, and Blue color values.

Range: 0 to 255

Default Value: 175, 143, 47

Syntax Example: `:DISP:COL:NORM:TRAC1:MEM 100, 0, 100`

`:DISP:COL:NORM:TRAC1:MEM?`

:DISPlay:COLor:RESet

Description: Resets all colors and inverted colors to their normal default values. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:DISP:COL:RES`

:DISPlay:COUNt <NRf>
:DISPlay:COUNt?

Description: Sets the number of displayed channels. When the VNA is in 25,000 point mode, the number of channels can only be 1 (one), 2, 3, 4, 6, 8, 9, 10, 12, or 16 channels. If the channel display is set to a non-listed number (5, 7, 11, 13, 14, 15), the instrument is set to the next higher channel number. If a number of greater than 16 is entered, the instrument is set to 16 channels. If the instrument is set to 100,000 points, any input results in 1 (one) channel. Outputs the number of displayed channels.

The example below assumes that the instrument is set to 25,000 points.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: For 25,000 point mode, 1 to 16 channels.

For 100,000 point mode, 1 channel.

Default Value: 1

Syntax Example: `:DISP:COUN 4`

`:DISP:COUN?`

:DISPlay:FSIGn[:STATe] <char>
:DISPlay:FSIGn[:STATe]?

Description: Turns on/off indicating a limit failure with a failure sign on the VNA screen. Outputs the on/off status of indicating a limit failure with a failure sign on the VNA screen.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:DISP:FSIG ON`

`:DISP:FSIG?`

:DISPlay:SIZe <char>
:DISPlay:SIZe?

Description: The command sets the maximum/normal size of the graticule display. The query outputs the maximum/normal size of the graticule display.

Cmd Parameters: <char> MAXimimum | NORMal

Query Parameters: <char> MAX | NORM

Range: NA

Default: NA

Syntax Example: `:DISP:SIZ`

`:DISP:SIZ?`

:DISPlay:SPLit <char>
:DISPlay:SPLit?

Description: Sets the channel display layout in a Row-by-Column format where the following display window arrangements are available:

- R = rows
- C = columns
- R1C1 = One trace on one row and one column
- R1C2 = Two traces, two across
- R2C1 = Two traces, two down
- R1C3 = Three traces, three across
- R3C1 = Three traces, three down
- R2C2C1 = Three traces, two on top, one on bottom
- R2C1C2 = Three traces, one on top, two on bottom
- C2R2R1 = Three traces, two on left, one on right
- C2R1R2 = Three traces, one on left, two on right
- R1C4 = Four traces, four across
- R4C1 = Four traces, four down
- R2C2 = Four traces, two across, two down
- R2C3 = Six traces, three across, two down
- R3C2 = Six traces, three down, two across
- R2C4 = Eight traces, four across, two down
- R4C2 = Eight traces, two across, four down
- R3C3 = Nine traces, three across, three down
- R5C2 = 10 traces, two across, five down
- R2C5 = 10 traces, five across, two down
- R4C3 = 12 traces, four across, three down
- R3C4 = 12 traces, three across, four down
- R4C4 = 16 traces, four across, four down

The query outputs channel display layout code as above. For additional descriptions, see the VectorStar operation manual.

Cmd Parameters: <char> R1C1 | R1C2 | R2C1 | R1C3 | R3C1 | R2C2C1 | R2C1C2 | C2R2R1 | C2R1R2 | R1C4 | R4C1 | R2C2 | R2C3 | R3C2 | R2C4 | R4C2 | R3C3 | R5C2 | R2C5 | R4C3 | R3C4 | R4C4

Query Parameters: <char> R1C1 | R1C2 | R2C1 | R1C3 | R3C1 | R2C2C1 | R2C1C2 | C2R2R1 | C2R1R2 | R1C4 | R4C1 | R2C2 | R2C3 | R3C2 | R2C4 | R4C2 | R3C3 | R5C2 | R2C5 | R4C3 | R3C4 | R4C4

Range: NA

Default Value: R1C1

Syntax Example: `:DISP:SPL R1C1`

`:DISP:SPL?`

:DISPlay:WINDow:ACTivate?

Description: Query only. Outputs the Active Channel number.

To specify an active channel, use the following command:

- `:DISPlay:WINDow{1-16}:ACTivate`

Query Parameters: NA

Syntax Example: `:DISP:WIND:ACT?`

:DISPlay:WINDow:Y:NDIVisions <NRf>
:DISPlay:WINDow:Y:NDIVisions?

Description: Enters the number of vertical divisions in the rectilinear displays of the active channel. Outputs the number of vertical divisions in the rectilinear displays of the active channel. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 4 to 30

Default Value: 10

Syntax Example: `:DISP:WIND:Y:NDIV 4`

`:DISP:WIND:Y:NDIV?`

:DISPlay:WINDow{1-16}:ACTivate

Description: The command sets the active channel to the indicated number. When the VNA is set to 100,000 point mode, the number of channels is limited to one. If the index number is omitted, the command activates Channel 1.

No query. To query about the active channel, use this command:

- `:DISPlay:WINDow:ACTivate?`

Cmd Parameters: <integer>

Range: 1 to 16

Default Value: 1

Syntax Example: `:DISP:WIND1:ACT`

:DISPlay:WINDow{1-16}:SIZe <char>
:DISPlay:WINDow{1-16}:SIZe?

Description: The command sets the maximum/normal size of the graphic display for the indicated channel.

The query returns the size of the graphic display for the indicated channel.

Cmd Parameters: < char> MAX | NORM

Query Parameters: NA

Query Output: <char> MAX | NORM

Range: NA

Default Value: NORM

Syntax Example: :DISP:WIND:SIZ MAX

:DISP:WIND:SIZ?

:DISPlay:WINDow{1-16}:SPLit <char>
:DISPlay:WINDow{1-16}:SPLit?

Description: Sets the trace display layout in a Row-by-Column format for the indicated channel. If more traces are set than the trace display contains, the higher numbered trace display windows are blank. If the trace display layout is less than the number of traces set, some traces will have overlapped displays. The following trace display window arrangements are available:

- R = rows
- C = columns
- R1C1 = One trace on one row and one column
- R1C2 = Two traces, two across
- R2C1 = Two traces, two down
- R1C3 = Three traces, three across
- R3C1 = Three traces, three down
- R2C2C1 = Three traces, two on top, one on bottom
- R2C1C2 = Three traces, one on top, two on bottom
- C2R2R1 = Three traces, two on left, one on right
- C2R1R2 = Three traces, one on left, two on right
- R1C4 = Four traces, four across
- R4C1 = Four traces, four down
- R2C2 = Four traces, two across, two down
- R2C3 = Six traces, three across, two down
- R3C2 = Six traces, three down, two across
- R2C4 = Eight traces, four across, two down
- R4C2 = Eight traces, two across, four down
- R3C3 = Nine traces, three across, three down
- R5C2 = 10 traces, two across, five down
- R2C5 = 10 traces, five across, two down
- R4C3 = 12 traces, four across, three down
- R3C4 = 12 traces, three across, four down
- R4C4 = 16 traces, four across, four down

The query outputs the trace display layout. For additional descriptions, see the VectorStar MS4640B VNA User Interface Reference Manual – 10410-00319.

Cmd Parameters: <char> R1C1 | R1C2 | R2C1 | R1C3 | R3C1 | R2C2C1 | R2C1C2 | C2R2R1 | C2R1R2 | R1C4 | R4C1 | R2C2 | R2C3 | R3C2 | R2C4 | R4C2 | R3C3 | R5C2 | R2C5 | R4C3 | R3C4 | R4C4

Query Parameters: <char> R1C1 | R1C2 | R2C1 | R1C3 | R3C1 | R2C2C1 | R2C1C2 | C2R2R1 | C2R1R2 | R1C4 | R4C1 | R2C2 | R2C3 | R3C2 | R2C4 | R4C2 | R3C3 | R5C2 | R2C5 | R4C3 | R3C4 | R4C4

Range: NA

Default Value: R2C2

Syntax Example: `:DISP:WIND1:SPL R1C1`

`:DISP:WIND1:SPL?`

:DISPlay:WINDow{1-16}:TITLe <string>
:DISPlay:WINDow{1-16}:TITLe?

Description: Sets the user title for the channel indicated. Outputs the user title for the channel indicated

Cmd Parameters: <string>

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:DISP:WIND1:TITL "Channel Title String"`

`:DISP:WIND1:TITL?`

:DISPlay:WINDow{1-16}:TITLe:STATe <char>
:DISPlay:WINDow{1-16}:TITLe:STATe?

Description: Enables/disables the display of the user title for the channel indicated. Outputs the enable/disable status of the user title display for the channel indicated.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:DISP:WIND1:TITL:STAT 1`

`:DISP:WIND1:TITL:STAT?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:SIZe <char>
:DISPlay:WINDow{1-16}:TRACe{1-16}:SIZe?

Description: Sets the maximum/normal size of the indicated trace on the indicated channel. Outputs the maximum/normal size of the indicated trace on the indicated channel

Cmd Parameters: <char> MAXimimum | NORMal

Query Parameters: <char> MAX | NORM

Range: NA

Default Value: NORMal

Syntax Example: `:DISP:WIND1:TRAC1:SIZ MAX`

`:DISP:WIND1:TRAC1:SIZ?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:X <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:X?

Description: The command sets X coordinate of the center for the Smith chart on the selected trace.

The query outputs the X coordinate of the center of the Smith chart on the selected trace.

Cmd Parameters: <NRf> The desired X coordinate for the selected trace (May be in exponential or non-exponential form.)

Query Parameters: NA

Query Output: <NR3> The current X coordinate of the selected trace in exponential form

Range: Dependent on the specific readout type selected:

Lin: 0 to 100

Log: –40 to 40

Real : –100 to 100

Re(Z): –10000 to 10000

Re(Y): –10000 to 10000

Default Value: 0 for most readout types

Syntax Example: `:DISP:WIND1:TRAC1:SMIT:CENT:X 5`

`:DISP:WIND1:TRAC1:SMIT:CENT:X?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:Y <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:Y?

Description: The command sets Y coordinate of the center for the Smith chart on the selected trace.

The query outputs the Y coordinate of the center of the Smith chart on the selected trace.

Cmd Parameters: <NRf> The desired Y coordinate for the selected trace (May be in exponential or non-exponential form.)

Query Parameters: NA

Query Output: <NR3> The current Y coordinate of the selected trace in exponential form

Range: Dependent on the specific readout type selected.

Phase: –180 to 180

Imag: –100 to 100

Imag(Z): –10000 to 10000

Imag(Y): –10000 to 10000

Default Value: 0 for most readout types

Syntax Example: `:DISP:WIND1:TRAC1:SMIT:CENT:Y -2`

`:DISP:WIND1:TRAC1:SMIT:CENT:Y?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:AUTO

Description: Auto scales the display of the indicated channel and trace. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:DISP:WIND1:TRAC1:Y:AUTO`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV?

Description: The command enters the per-division scale value of the top display of the indicated channel and trace subject to the limitations described below. The query outputs the per-division scale value of the top display of the indicated channel and trace. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Working with Dual-Display Rectangular Trace Formats

This command is primarily designed to be used to set the per-division (PDIV) scale value on one of the VectorStar VNA dual-display trace formats. There are four dual-display trace formats:

- Linear Magnitude and Phase (LINPHase)
- Log Magnitude and Phase (LOGPHase)
- Real and Imaginary (REIMaginary)
- Impedance Real and Imaginary (ZCOMPlex)

Working with Single-Display Rectangular or Polar Trace Formats

If the trace is a single-trace non-Smith Chart display, this command can also set the per-division scale value on single trace displays including rectangular and polar graph trace formats.

Working with Smith Chart Trace Displays

If the trace type is a Smith Impedance Chart or a Smith Admittance Chart, the available scale values are limited to the values of +3dB, 0dB, -10dB, -20dB, -30dB. These are the only values permitted and other entered values result in an execution error.

Cmd Parameters: For Smith Charts: <NRf> 3 | 0 | -10 | -20 | -30

Where:

- 3 = +3dB compressed Smith Chart
- 0 = 0 dB standard Smith Chart
- -10 = -10 dB expanded Smith Chart
- -20 = -20 dB expanded Smith Chart
- -30 = -30 dB expanded Smith Chart

For all other displays: <NRf>

Query Parameters: <NR3> The output parameter units varies depending on the trace display type.

For Smith Charts, the output parameter is based on the outside radius of the outer circle. For all other trace displays, the output is in one of the following:

- dB per division
- Hertz per division
- Meters per division
- Seconds per division.

Range: The range varies depending on display type:

- For Log Magnitude display types: 1E-3 to 1E3
- For all other non-Smith display types: 1E-5 to 1E9
- For all Smith Chart display types, discrete values only: 3 | 0 | -10 | -20 | -30.

Default Value: 1.000000E+001

Syntax Example: `:DISP:WIND1:TRAC1:Y:PDIV 1E2`

`:DISP:WIND1:TRAC1:Y:PDIV?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV2 <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV2?

Description: The command enters the per-division scale value of the bottom display of the indicated channel and trace. It does not create an error if the command is invoked using a one-trace display, but, when a two-trace dis lay is re-invoked, all changes made to the “hidden” trace are discarded and the trace reverts to its prior visible settings. The query outputs the per-division scale value of the bottom display of the indicated channel and trace.

See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: 1E-5 to 1E9

Default Value: 1.000000E+001

Syntax Example: `:DISP:WIND1:TRAC1:Y:PDIV2 5E0`

`:DISP:WIND1:TRAC1:Y:PDIV2?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:OFFSet <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:OFFSet?

Description: Enters the phase offset value for the display of the indicated channel and trace. Outputs the phase offset value for the display of the indicated channel and trace. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter is in Degrees.

Query Parameters: <NR3> The output parameter is in Degrees.

Range: -3.6E2 to 3.6E2

Default Value: 0.000000E+000

Syntax Example: `:DISP:WIND1:TRAC1:Y:PHAS:OFFS 4.5E1`

`:DISP:WIND1:TRAC1:Y:PHAS:OFFS?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:WRAPping[:STATe] <char>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:WRAPping[:STATe]?

Description: The command turns phase wrapping on/off on the indicated trace and channel. This is only used with rectangular graph trace displays. The query outputs the on/off status of Phase Wrapping on the indicated trace and channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:DISP:WIND1:TRAC1:Y:PHAS:WRAP 1`

`:DISP:WIND1:TRAC1:Y:PHAS:WRAP?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHOFF <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHOFF?

Description: The command enters the phase offset value for the display of the indicated trace and channel. This is only used with rectangular graph trace displays. The query outputs the phase offset value for the display of the indicated trace and channel.

Cmd Parameters: <NRf> The input parameter is in Degrees.

Query Parameters: <NR3> The output parameter is in Degrees.

Range: -3.6E2 to 3.6E2

Default Value: 0.00000000000E+000

Syntax Example: `:DISP:WIND1:TRAC1:Y:PHOFF 2.10E1`

`:DISP:WIND1:TRAC1:Y:PHOFF?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV?

Description: Enters the reference level of the top display of the indicated channel and trace. Outputs the reference level of the top display of the indicated channel and trace.

See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: -9.999E2 to +9.999E2

Default Value: 0.00

Syntax Example: `:DISP:WIND1:TRAC1:Y:RLEV -6.0E1`

`:DISP:WIND1:TRAC1:Y:RLEV?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV2 <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV2?

Description: Enters the reference level of the bottom display of the indicated channel and trace. Outputs the reference level of the bottom display of the indicated channel and trace. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: -9.999E2 to +9.999E2

Default Value: 0.00

Syntax Example: `:DISP:WIND1:TRAC1:Y:RLEV2 -6.0E1`

`:DISP:WIND1:TRAC1:Y:RLEV2?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS?

Description: Enters the reference line position of the top display of the indicated channel and trace. This command also works with single-display traces. Outputs the reference line position of the top display of the indicated channel and trace. See Table 2-7, "Trace Parameters and Coefficients" on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: Minimum Reference Line Position = 0

Maximum Reference Line Position = Maximum Number of Divisions set by :DISPlay:Y:NDIVisions <NRf>

Default Value: 5

Syntax Example: `:DISP:WIND1:TRAC1:Y:RPOS 10`

`:DISP:WIND1:TRAC1:Y:RPOS?`

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS2 <NRf>
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS2?

Description: For dual-trace displays only. The command enters the reference line position of the bottom display of the indicated channel and trace. If the trace display is a single-trace display, this command results in an execution error. Outputs the reference line position of the bottom display of the indicated channel and trace. See Table 2-7, "Trace Parameters and Coefficients" on page 2-22 for a complete listing of trace graph types, default settings, and available ranges.

Cmd Parameters: <NRf> The input parameter depends on the display type.

Query Parameters: <NR3> The output parameter depends on the display type.

Range: Minimum Reference Line Position = 0

Maximum Reference Line Position = Maximum Number of Divisions set by :DISPlay:Y:NDIVisions <NRf>

Default Value: 5

Syntax Example: `:DISP:WIND1:TRAC1:Y:RPOS2 10`

`:DISP:WIND1:TRAC1:Y:RPOS2?`

Related Cmds: ":DISPlay:Y:NDIVisions" on page 5-241

:DISPlay:WINDow{1-16}:Y:AUTO

Description: Auto scales all traces of the indicated channel. See Table 2-7, "Trace Parameters and Coefficients" on page 2-22 for a complete listing of trace graph types, default settings, and available ranges. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:DISP:WIND1:Y:AUTO`

:DISPlay:Y:AUTO

Description: Auto scales all traces on all channels. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:DISP:Y:AUTO`

:DISPlay:Y:NDIVisions

Description: Programs the number of vertical divisions in the active channel into all rectilinear displays. See Table 2-7, “Trace Parameters and Coefficients” on page 2-22 for a complete listing of trace graph types, default settings, and available ranges. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:DISP:Y:NDIV`

5-51 :FORMat Subsystem

The :FORMat subsystem commands are used on a per-instrument basis to configure, control, and query the format for I/O data.

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- ":CALCulate{1-16}:FORMat Subsystem - SnP Data" on page 5-49
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":FORMat Subsystem" on page 5-242
- ":HCOPy Subsystem" on page 5-245
- ":MMEMory Subsystem" on page 5-251

:FORMat:BORDer <char>
:FORMat:BORDer?

Description: Sets the binary numeric I/O data byte order as either Most Significant Byte (MSB) or Least Significant Byte (LSB) byte order of binary numeric I/O data where:

- NORMal = The MSB is first
- SWAPped = The LSB is first.

The query outputs the Most Significant Byte (MSB)/Least Significant Byte (LSB) byte order of binary numeric I/O data.

Cmd Parameters: <char> NORMal | SWAPped

Query Parameters: <char> NORM | SWAP

Range: NA

Default Value: SWAP

Syntax Example: `:FORM:BORD NORM`

`:FORM:BORD?`

:FORMat:DATA <char>
:FORMat:DATA?

Description: The command sets the format for numeric I/O data representation where:

- ASCii = An ASCII number of 20 or 21 characters long with floating point notation. For example, 12345E-4.
- REAL = 8 Bytes of binary floating point number representation limited to 64 bits which is the value of the MPND (Maximum Positive/Negative Double Precision Number or ± 1.792 631 348 6E38)
- REAL32 = 4 Bytes of floating point number representation which is the value of the MPNF (Maximum Positive/Negative Floating Point Number or ± 3.402 819E38)

The query outputs the format of numeric I/O data representation.

Cmd Parameters: <char> ASCii | REAL | REAL32

Query Parameters: <char> ASC | REAL | REAL32

Range: NA

Default Value: ASC

Syntax Example: `:FORM:DATA REAL`

`:FORM:DATA?`

:FORMat:DATA:HEADing[:STATe] <char>
:FORMat:DATA:HEADing[:STATe]?

Description: Enables or disables including a heading with data files. Outputs the enable/disable status of including a heading with data files.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:FORM:DATA:HEAD ON`

`:FORM:DATA:HEAD?`

:FORMat:SNP:FREQuency <char>
:FORMat:SNP:FREQuency?

Description: Sets the frequency unit displayed in an SNP data file. Queries the frequency unit displayed in an SNP data file.

Cmd Parameters: <char> HZ | KHZ | MHZ | GHZ

Query Parameters: <char> HZ | KHZ | MHZ | GHZ

Range: NA

Default Value: GHZ

Syntax Example: `:FORM:SNP:FREQ HZ`

`:FORM:SNP:FREQ?`

:FORMat:SNP:PARameter <char>
:FORMat:SNP:PARameter?

Description: Sets the parameter format displayed in an SNP data file. Outputs the parameter format displayed in an SNP data file.

Cmd Parameters: <char> LINPH | LOGPH | REIM

Where:

- LINPH = Linear and Phase
- LOGPH = Log and Phase
- REIM = Real and Imaginary Numbers

Query Parameters: <char> LINPH | LOGPH | REIM

Range: NA

Default Value: REIM

Syntax Example: `:FORM:SNP:PAR LINPH`

`:FORM:SNP:PAR?`

5-52 :HCOPy Subsystem

The :HCOPy subsystem commands are used to create print output default settings for the instrument graphics display.

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- “:CALCulate{1-16}:FORMat Subsystem - SnP Data” on page 5-49
- “:CALCulate{1-16}:NXN Subsystem” on page 5-88
- “:CALCulate{1-16}[:SELected]:DATA Subsystem” on page 5-136
- “:CALCulate{1-16}[:SELected]:TDATA Subsystem” on page 5-197
- “:FORMat Subsystem” on page 5-242
- “:HCOPy Subsystem” on page 5-245
- “:MMEMory Subsystem” on page 5-251

:HCOPy:DEVice:ID <String>
:HCOPy:DEVice:ID?

Description: The command enters the Device Identity string for the header printout. The query outputs the Device Identify string for the header printout.

Cmd Parameters: <String>

Query Parameters: NA

Output: <char>

Range: NA

Default: NA

Syntax Example: `:HCOP:DEV:ID <char>`

`:HCOP:DEV:ID?`

:HCOPy:DEVice:ID:STATe <char>
:HCOPy:DEVice:ID:STATe?

Description: The command enters the on/off state of the Device Identity string for the header printout. The query outputs the on/off state of the Device Identify string for the header printout.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Output: <char> 1 | 0

Range: NA

Default: NA

Syntax Example: `:HCOP:DEV:ID:STAT <char>`

`:HCOP:DEV:ID:STAT?`

:HCOPy:IMAGe <char>
:HCOPy:IMAGe?

Description: Sets the hardcopy print color for the display where:

- NORMal = As the instrument screen is displayed.
- INVert = Inverts the colors. Typically used to print the traces as colors, and the instrument display background as white.
- BWHITe = Converts all colors to either black or white.

Outputs the hardcopy print color for the display.

Cmd Parameters: <char> NORMal | INVert | BWHITe

Query Parameters: <char> NORM | INV | BWHIT

Range: NA

Default Value: INV

Syntax Example: `:HCOP:IMAG NORM`

`:HCOPy:IMAGe?`

:HCOPy:MODel <String>
:HCOPy:MODel?

Description: Enter the Model string for the header printout. Output the Model string for the header printout

Cmd Parameters: <String>

Query Parameters: NA

Output: <char>

Range: NA

Default: NA

Syntax Example: `:HCOP:MOD <String>`

`:HCOP:MOD?`

:HCOPy:MODel:STATe<char>
:HCOPy:MODel:STATe?

Description: Enter the on/off state of the Model string for the header printout. Output the on/off state of the Model string for the header printout.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Output: <char> 1|0

Range: NA

Default: NA

Syntax Example: `:HCOP:MOD:STAT <char>`

`:HCOP:MOD:STAT?`

:HCOPy:OPERator:COMMent<String>
:HCOPy:OPERator:COMMent?

Description: Enter the Operator Comment string for the header printout. Output the Operator Command string for the header printout.

Cmd Parameters: <String>

Query Parameters: NA

Output: <char>

Range: NA

Default: NA

Syntax Example: `:HCOP:OPER:COMM <String>`

`:HCOP:OPER:COMM?`

:HCOPy:OPERator:COMMent:STATe <char>
:HCOPy:OPERator:COMMent:STATe?

Description: Enters the on/off state of Enter the Operator Comment string for the header printout. Output the on/off state of the Operator Command string for the header printout.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Output: <char> 1|0

Range: NA

Default: NA

Syntax Example: `:HCOP:OPER:COMM:STAT <char>`

`:HCOP:OPER:COMM:STAT?`

:HCOPy:OPERator:NAMe <String>
:HCOPy:OPERator:NAMe?

Description: Enter the Operator Name string for the header printout. Output the Operator Name string for the header printout.

Cmd Parameters: <String>

Query Parameters: NA

Output: <char>

Range: NA

Default: NA

Syntax Example: `:HCOP:OPER:NAM <String>`

`:HCOP:OPER:NAM?`

:HCOPy:OPERator:NAMe:STATe <char>
:HCOPy:OPERator:NAMe:STATe?

Description: Enter the on/off state of the Operator Name string for the header printout. Output the on/off state of the Operator Name string for the header printout.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Output: <char> 1|0

Range: NA

Default: NA

Syntax Example: :HCOP:OPER:NAM:STAT <char>

:HCOP:OPER:NAM:STAT?

:HCOPy:PRINt:DATe:TIMe:STATe <char>
:HCOPy:PRINt:DATe:TIMe:STATe?

Description: Turns on/off printing the Date Time information. Output the on/off state of printing the Date Time information.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Output: <char> 1|0

Range: NA

Default: NA

Syntax Example: :HCOP:PRIN:DAT:TIM:STAT <char>

:HCOP:PRIN:DAT:TIM:STAT?

:HCOPy:PRINt:HEADers:STATe <char>
:HCOPy:PRINt:HEADers:STATe?

Description: Turns on/off printing the header information. Output the on/off state of printing the header information.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Output: <char> 1|0

Range: NA

Default: NA

Syntax Example: :HCOP:PRIN:HEAD:STAT <char>

:HCOP:PRIN:HEAD:STAT?

:HCOPy:PRINt:LOGO:STATe <char>
:HCOPy:PRINt:LOGO:STATe?

Description: Turns on/off printing a Logo bitmap at the top of the print page. Output the on/off state of printing a Logo bitmap at the top of the print page.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Output: <char> 1|0

Range: NA

Default: NA

Syntax Example: `:HCOP:PRIN:LOGO:STAT <char>`

`:HCOP:PRIN:LOGO:STAT?`

:HCOPy:PRINt:LOGO:TYPe <char>
:HCOPy:PRINt:LOGO:TYPe?

Description: Sets the logo type to print at the top of the print page. Queries the logo type to print at the top of the print page.

Cmd Parameters: <char> ANRitsu|USER

Where:

- ANRitsu = standard Anritsu corporate logo.
- USER = user defined logo.

Query Parameters: NA

Output: <char> ANR | USER

Range: NA

Default: NA

Syntax Example: `:HCOP:PRIN:LOGO:TYP <char>`

`:HCOP:PRIN:LOGO:TYP?`

:HCOPy:PRINt:TYPe <char>
:HCOPy:PRINt:TYPe?

Description: Select the hardcopy print type for the HCOPy print command. Output the hardcopy print type selected for the HCOPy command.

Cmd Parameters: <char> BITMap|GRAPhical|TABular

Query Parameters: NA

Output: <char> BITM |GRAP |TAB

Range: NA

Default: NA

Syntax Example: `:HCOP:PRIN:TYP <char>`

`:HCOP:PRIN:TYP?`

:HCOPy:USER:LOGO
:HCOPy:USER:LOGO?

Description: Enter the full filespec for the desired user logo. Output the resultant full filespec for the user logo.

Cmd Parameters: <String>

Query Parameters: NA

Output: <char>

Range: NA

Default: NA

Syntax Example: `:HCOP:USER:LOGO <String>`

`:HCOP:USER:LOGO?`

:HCOPy[:IMMediate]

Description: Prints the display image to the default printer. The default printer is set through the standard Windows PRINT SETUP dialog box.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:HCOP`

5-53 :MMEMory Subsystem

The :MMEMory subsystem commands are used to input, load, and read out instrument data files. <String> formatted data is generally used to represent file directories and file names.

I/O Configuration and File Operation Subsystems

Related subsystems for I/O configuration and file operation are:

- ":CALCulate{1-16}:FORMat Subsystem - SnP Data" on page 5-49
- ":CALCulate{1-16}:NXN Subsystem" on page 5-88
- ":CALCulate{1-16}[:SELected]:DATA Subsystem" on page 5-136
- ":CALCulate{1-16}[:SELected]:TDATA Subsystem" on page 5-197
- ":FORMat Subsystem" on page 5-242
- ":HCOPy Subsystem" on page 5-245
- ":MMEMory Subsystem" on page 5-251

:MMEMory:CATalog? {<string>}

Description: Query only. Read out the directory information of the storage device. If the directory does not exist, it results in an execution error.

Query Parameters: <string> Path specification in the form: 'x:\directory' where x: must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:CAT? 'C:\directory'`

:MMEMory:COPY <string1>, <string2>

Description: The command copies the contents of the first <string1> file to the second <string2> file. No query. The directory and file for <string1> must exist. If the directory and file for <string2> do not exist, they will be created. If the directory and file for <string1> already exist, they will be overwritten.

Cmd Parameters: <string1> Filename and path in the form: 'x:\directory\filename.xxx' where x:\directory\filename.xxx must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:COPY 'C:\filename1.s2p', 'C:\filename2.s2p'`

:MMEMory:DELete <string>

Description: Delete a disk, file, or directory. Use caution with this command as there is no recovery operation in case of a user mistake or error. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.xxx' or 'x:\directory' where x:\directory\filename.xxx or x:\directory must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:DEL 'C:\directory\filename.s2p'`

:MMEMory:LOAD <string>

Description: Loads data file whose type is specified by the filename extension into the instrument memory. The directory and file in <string> must exist. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.xxx' where x:\directory\filename.xxx must exist.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:LOAD 'C:\filename.s2p'`

:MMEMory:LOAD:CKIT <string>

Description: Loads a calibration kit file or multiple files from the given file specification into the instrument memory. The :MMEMory:LOAD:CKIT command supports loading XML files with a .ccf extension or binary files in Lightning format. No query.

Cmd Parameters: <string> Filename and path where the required format changes depending on the file type being loaded.

If a .ccf file type is used, the required form is: 'x:\directory\filename.ccf' where x:\directory\filename.ccf must exist.

If a binary Lightning file type is used, only the path is required in the form: 'x:\directory' where x:\directory and cal kit files must exist.

See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: For calibration kit or other .ccf files:

```
:MMEM:LOAD:CKIT 'E:\calibrationfoldermykit.ccf'
```

For binary files:

```
:MMEM:LOAD:CKIT 'E:\calibrationbinaries'
```

:MMEMory:LOAD:FLAT <string>

Description: Loads the flat test port power flatness coefficients to the active channel in the form of an .fpc file. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.fpc' where x:\directory\filename.fpc must exist. See definition of "<string>" on page 2-12 and Table 2-8 for a list of supported file types.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:LOAD:FLAT 'C:\pwrflt.fpc'`

:MMEMory:LOAD:FSEGMent <string>

Description: Loads the frequency-based segmented sweep table from the given filespec to the active channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.sgs' where x:\directory\filename.sgs must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:LOAD:FSEGM 'C:\directory\filename.sgs'`

:MMEMory:LOAD:ISEGMent <string>

Description: Loads the index-based segmented sweep table from the given file specification to the active channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.sgs' where x:\directory\filename.sgs must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:LOAD:ISEGM 'C:\directory\filename.sgs'`

:MMEMory:LOAD:LIMit <string>

Description: Loads a Limit Table from the file system to the active channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.lmt' where x:\directory\filename.lmt must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:LOAD:LIM 'C:\limits3.lmt'`

:MMEMory:LOAD:LINearity <string>

Description: Loads the power sweep linearity coefficients to the active channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.ppc' where x:\directory\filename.ppc must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:LOAD:LIN 'C:\linch3.ppc'`

:MMEMory:LOAD:MDATA <string>

Description: Loads formatted or unformatted trace data from a file into the trace memory of the active trace on the active channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.tdf' or 'x:\directory\filename.tdu' where x:\directory\filename.tdf or x:\directory\filename.tdf must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:LOAD:MDATA 'C:\directory\filename.tdu'`

:MMEMory:MDIRectory <string>

Description: Create a new disk directory. No query.

Cmd Parameters: <string> Path specification in the form: 'x:\directory' where x: must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:MDIR 'C:\directory'`

:MMEMory:RDIRectory <string>

Description: Delete a disk directory. No query.

Cmd Parameters: <string> Path specification in the form: 'x:\directory' where x:\ and \directory\ must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:RDIR 'C:\directory'`

:MMEMory:STORe <string>

Description: Stores a data file of the type specified by the filename extension.No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.xxx' where x:\directory\ must exist. See definition of "<string>" on page 2-12 and Table 2-8 for a list of supported file types.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR 'C:\filename.acd'`

:MMEMory:STORe:FLAT{1-7} <string>

Description: Saves the flat test port power flatness coefficients for the indicated port and the active channel. No query. For port definitions, see Table 5-1.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.fpc' where x:\directory\ must exist. See definition of "<string>" on page 2-12 and Table 2-8 for a list of supported file types.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR:FLAT1 'C:\filename.fpc'`

Table 5-1. Port Number Definitions (1 of 2)

Port Number	Definition
1	Port1 (2 and 4 port system)
2	Port2 (2 and 4 port system)
3	Src2 out Port1 (2 port system with option 32), (or Port3 on a 4 port system)
4	Port4 (4 port system)
5	Src2 out Port1 (4 port system)

Table 5-1. Port Number Definitions (2 of 2)

Port Number	Definition
6	Src2 out Port2 (4 port system)
7	Src1 driving Aux Mod (ME7838x System)

:MMEMory:STORe:FSEGMent <string>

Description: Stores the frequency-based segmented sweep table of the active channel to the given filespec. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.sgs' where x:\directory\ must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR:FSEGM 'C:\directory\filename.sgs'`

:MMEMory:STORe:IMAGe <string>

Description: Save the display image to a disk file (PNG or BMP format). No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.bmp' where x:\directory\ must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR:IMAG 'C:\directory\filename.bmp'`

:MMEMory:STORe:ISEGMent <string>

Description: Stores the index-based segmented sweep table of the active channel to the given filespec. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.sgs' where x:\directory\ must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR:ISEGM 'C:\directory\filename.sgs'`

:MMEMory:STORe:LIMit <string>

Description: Stores the limit table of the active channel to the hard disk.No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.lmt' where x:\directory must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR:LIM 'C:\directory\filename.lmt'`

:MMEMory:STORe:LINearity{1-2} <string>

Description: Saves the power sweep linearity coefficients for the indicated port and the active channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.ppc' where x:\directory must exist. See definition of "<string>" on page 2-12 and Table 2-8, "Supported File Types" on page 2-26 for a list of supported file types.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR:LIN1 'C:\directory\filename.ppc'`

:MMEMory:STORe:MDATA <string>

Description: Save formatted or unformatted trace memory data to the hard disk from the active trace on the active channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.tdf' or 'x:\directory\filename.tdu' where x:\directory must exist. See definition of "<string>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:STOR:MDATA 'C:\directory\filename.tdu'`

:MMEMory:STORe:MP41 <string>

Description: Store M4P data file using a one Differential pair with two Singleton mixed mode model. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.m4p' where x:\directory must exist. See definition of "<string>" on page 2-12.

Range: NA

Default: NA

Syntax Example: `:MMEM:STOR:MP41 "C:AnritsuVNAmym4pfile.m4p"`

:MMEMory:STORe:MP42 <string>

Description: Store M4P data file using the two Differential Pairs and zero Singleton mixed mode model. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.m4p' where x:\directory must exist. See definition of "<string>" on page 2-12.

Range: NA

Default: NA

Syntax Example: `:MMEM:STOR:MP42 "C:AnritsuVNAmym4pfile.m4p"`

:MMEMory:TRANsfer <string>, <block>
:MMEMory:TRANsfer?

Description: The command writes GPIB data to a disk file. The query outputs the disk file data to the GPIB. The file must exist.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.xxx' where x:\directory must exist.

<block> Binary data block must exist.

Query Parameters: <string> The filename and path in the form: 'x:\directory\filename.xxx' where x:\directory\filename.xxx must exist.

See definition of "<string>" on page 2-12 and "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:MMEM:TRAN 'C:\directory\filename.cal', <block>`

`:MMEM:TRAN? 'C:\directory\filename.cal'`

:MMEMory:WRITE:CKIT <char1>, <char2>, <char3>, <string>

Description: This command is used to write cal kits for particular connectors and waveguide sizes.The command writes a calibration kit file from current setup. All three <char> parameters must be specified. The <string> parameter defines either the directory and file name or only the file name subject to the limitations described below. No query.

Required Parameters

Three (3) identifying characteristics must be provided as Line Type, Cal Type, and one of three Connector Types (Coaxial, Waveguide, User Defined). Each parameter characteristic is defined below:

- Line Type = <char1> = COAXial | NONDispersive | WAVEguide. Note that Microstrip is not an available option.
- Cal Type = <char2> = SOLX | SSLT | SSST
- Connector/Waveguide Type = <char3> = Coaxial type connectors, waveguide types, or user defined connectors and waveguide as listed below.
- Coaxial Connector Types = <char3> = CID2-V | CID2 | CID3 | CIDK | CIDN75 | CIDN | CIDS | CIDT | CIDT | CIDV | SID1 | CID0.8 | CID716 | CIDG
- Waveguide Types = <char3> = CIDWR10 | CIDWR12 | CIDWR15
- User Defined Connectors and Waveguides = <char3> = CIDU1 | CIDU2 | CIDU3 | CIDU4 | CIDU5 | CIDU6 | CIDU7 | CIDU8

For SOLR and SOLT calibration methods, use the SOLX parameter.

If the line type is coaxial and the calibration method is SSLT or SSST, the connector type must be CID1.

For waveguide line types, the calibration method must be SSLT or SSST.

See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their parameters.

Parameter Definitions for <char1>

<char> COAXial | NONDISpersive | WAVEguide

Parameter Definitions for <char2>

<char2> SOLX | SSLT | SSST, where:

- SOLX = Short-Open-Load-Reciprocal or Short-Open-Load-Through calibration
- SSLT = Short-Short-Load-Through
- SSST = Short-Short-Short-Through

Parameter Definitions for <char3>

<char3> CID1 | CIDV | CIDK | CID2 | CID2-V | CID3 | CIDS | CIDN | CIDN75 | CIDG | CID716 | CIDT | CID0.8 | CIDWR10 | CIDWR12 | CIDWR15 | CIDU1 | CIDU2 | CIDU3 | CIDU4 | CIDU5 | CIDU6 | CIDU7 | CIDU8 | CIDU9 | CIDU10 | CIDU11 | CIDU12 | CIDU13 | CIDU14 | CIDU15 | CIDU16 | CIDU17 | CIDU18 | CIDU19 | CIDU20 | CIDU21 | CIDU22 | CIDU23 | CIDU24 | CIDU25 | CIDU26 | CIDU27 | CIDU28 | CIDU29 | CIDU30 | CIDU31 | CIDU32, where:

CID1 = 1 mm	CIDU10 = User defined 10
CIDV = V	CIDU11 = User defined 11
CIDK = K	CIDU12 = User defined 12
CID2 = 2.4 mm	CIDU13 = User defined 13
CID2-V = Anritsu 2.4 mm	CIDU14 = User defined 14
CID3 = GPC-3.5	CIDU15 = User defined 15
CIDS = SMA	CIDU16 = User defined 16
CIDN = N	CIDU17 = User defined 17
CIDN75 = N-connector, 75 Ohm	CIDU18 = User defined 18
CIDG = GPC-7	CIDU19 = User defined 19
CID716 = 7/16 inch	CIDU20 = User defined 20
CIDT = TNC	CIDU21 = User defined 21
CID0.8 = 0.8 mm	CIDU22 = User defined 22
CIDWR10 = WR-10 Waveguide	CIDU23 = User defined 23
CIDWR12 = WR-12 Waveguide	CIDU24 = User defined 24
CIDWR15 = WR-15 Waveguide	CIDU25 = User defined 25
CIDU1 = User defined 1[a]	CIDU26 = User defined 26
CIDU2 = User defined 2	CIDU27 = User defined 27
CIDU3 = User defined 3	CIDU28 = User defined 28
CIDU4 = User defined 4	CIDU29 = User defined 29
CIDU6 = User defined 5	CIDU30 = User defined 30
CIDU7 = User defined 7	CIDU31 = User defined 31
CIDU8 = User defined 8	CIDU32 = User defined 32
CIDU9 = User defined 9	

a. Note that a user-defined CIDU1 connector can be assigned a user-defined name, but programmatically, it must be referred to as “CIDU1”. This also applies to CIDU2 through CIDU32.

Parameter Definitions for <string>

User-provided variations in the <string> parameter command suffix and existing O/S conditions change the way the command will operate.

- Best practices recommend stating a file name and path in the form 'x:\directory\filename.xxx' where 'x:\directory\' already exists.
- If the directory and file name in <string> already exist, the command will overwrite them without comment.
- If the directory portion of the <string> is missing such as 'x:\filename.xxx', the named file is placed in the current O/S defined default directory.
- If the command uses a <string> in the path and file name form such as 'x:\directory\filename.xxx' and 'x:\directory\' does not exist, an error is generated and no file is written.
- See definition of "<string>" on page 2-12.

Cmd Parameters: <char1> COAXial | NONDISpersive | WAVEguide

<char2> SOLX | SSLT | SSST

<char3> CID1 | CIDV | CIDK | CID2 | CID2-V | CID3 | CIDS | CIDN | CIDN75 | CIDG | CID716 | CIDT | CID0.8 | CIDWR10 | CIDWR12 | CIDWR15 | CIDU1 | CIDU2 | CIDU3 | CIDU4 | CIDU5 | CIDU6 | CIDU7 | CIDU8 | CIDU9 | CIDU10 | CIDU11 | CIDU12 | CIDU13 | CIDU14 | CIDU15 | CIDU16 | CIDU17 | CIDU18 | CIDU19 | CIDU20 | CIDU21 | CIDU22 | CIDU23 | CIDU24 | CIDU25 | CIDU26 | CIDU27 | CIDU28 | CIDU29 | CIDU30 | CIDU31 | CIDU32, where:

<string> Path and filename in the form 'x:\directory\filename.xxx' or 'x:\filename.xxx'. If the 'x:\directory\' form is used, the directory must exist. See <string> parameter definitions above and definition of "<string>" on page 2-12.

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:MMEM:WRITE:CKIT COAX, SOLX, CIDV, 'C:\directory\calfilename.ccf'`

5-54 :SENSe:BANDwidth Subsystem

The :SENSe:BANDwidth subsystem command toggles the IF enhancer state on and off on an per-instrument basis.

IF Configuration Subsystems

Related IF configuration and control subsystems are:

- ":SENSe:BANDwidth Subsystem" on page 5-260
- ":SENSe{1-16}:BANDwidth Subsystem" on page 5-267
- ":SENSe{1-16}:BWIDth Subsystem" on page 5-268

:SENSe:BANDwidth:ENHancer[:STATe] <char>
:SENSe:BANDwidth:ENHancer[:STATe]?

Description: Sets the on/off state of the IF bandwidth enhancer. Outputs the on/off state of the IF bandwidth enhancer.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS:BAND:ENH ON`

`:SENS:BAND:ENH?`

5-55 :SENSe:HOLD Subsystem

The :SENSe:HOLD subsystem command sets the hold function for all channels on a per-instrument basis.

Trigger, Hold, and External Source Subsystems

Related trigger, hold, and external source subsystems are:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419
- ":SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem" on page 5-452
- ":SENSe{1-16}:SOURce{1-4} Subsystem" on page 5-488
- ":SOURce:ALL:EXTernal Subsystem" on page 5-508
- ":SOURce{1-4}:EXTernal Subsystem" on page 5-526
- ":TRIGger[:SEQuence] Subsystem" on page 5-552

:SENSe:HOLD:FUNCtion <char>
:SENSe:HOLD:FUNCtion?

Description: Sets the hold function for all channels where the following hold options are available:

- CONTinuous = Perform continuous sweeps on all channels
- HOLD = Hold the sweep on all channels
- SINGle = Perform a single sweep and then hold all channels

Outputs the hold status for all channels.

The operation of this command depends on the settings of the :SENSe:HOLD:FUNCtion, :TRIGger[:SEQuence][:REMote]:SINGle, and :TRIGger[:SEQuence][:IMMediate][:REMote] commands. Each setting combination is described in the sections below.

:SENSe:HOLD:FUNC CONT and :TRIG

```
:SENSe:HOLD:FUNCtion CONTinuous

// Sweep State = The sweep is sweeping continuously

// Command Execution = The parser is ready for a command right
away.

:TRIGger[:SEQuence][:IMMediate][:REMote]

// Sweep State = The sweep restarts and sweeps continuously. When
the sweep gets to the end of the sweep, it continues to sweep.
There is NO STATUS information that the end of the sweep has been
reached.

// Command Execution = The parser is ready for a command right away
```

:SENSe:HOLD:FUNC CONT and :TRIG:SING

```
:SENSe:HOLD:FUNCtion CONTinuous
// Sweep State = The sweep is sweeping continuously.
// Command Execution = The parser is ready for a command right away
:TRIGger[:SEQuence][:REMote]:SINGle
// Sweep State = The sweep restarts and sweeps continuously. When
the sweep gets to the end of the sweep, it sets the end of sweep
status bit and continues to sweep.
// Command Execution = Further execution is blocked until the end
of the sweep.
// Command Execution resumes when the sweep has reached the end of
the sweep.
```

:SENSe:HOLD:FUNC HOLD and :TRIG

```
:SENSe:HOLD:FUNCtion HOLD
// Sweep State = The sweep is stopped.
// Command Execution = The parser is ready for a command right away
:TRIGger[:SEQuence][:IMMediate][:REMote]
// Sweep State = The command has no effect. The sweep is stopped.
// Command Execution = The parser is ready for a command right away
```

:SENSe:HOLD:FUNC HOLD and :TRIG:SING

```
:SENSe:HOLD:FUNCtion HOLD
// Sweep State = The sweep is stopped
// Command Execution = The parser is ready for a command right
away.
:TRIGger[:SEQuence][:REMote]:SINGle
// Sweep State = The sweep restarts and sweeps until the end of the
sweep, at which point it sets the end of sweep status bit and
stops.
// Command Execution = Further execution is blocked until the end
of the sweep.
// Command Execution resumes when the sweep has reached the end of
the sweep.
```

:SENSe:HOLD:FUNC SING and :TRIG

```
:SENSe:HOLD:FUNCtion SINGle
// Sweep State = The sweep does one complete sweep, goes into hold
and stops.
// Command Execution = The parser is ready for a command.
:TRIGger[:SEQuence][:IMMediate][:REMote]
// Sweep State = The command has no effect. The sweep is stopped.
// Command Execution = The parser is ready for a command.
```

:SENSe:HOLD:FUNC SING and :TRIG:SING

```
:SENSe:HOLD:FUNCtion SINGle
// Sweep State = The sweep does one complete sweep, goes into hold and stops.
// Command Execution = The parser is ready for a command right away.
:TRIGger[:SEQuence][:REMote]:SINGle
// Sweep State = The sweep restarts and sweeps until the end of the sweep, at which point it sets the end of sweep status bit and stops.
// Command Execution = Further execution is blocked until the end of the sweep.
// Command Execution resumes when the sweep has reached the end of the sweep.
```

Cmd Parameters: <char> CONTinuous | HOLD | SINGle

Query Parameters: <char> CONT | HOLD | SING

Range: NA

Default Value: NA

Syntax Example: `:SENS:HOLD:FUNC CONT`

`:SENS:HOLD:FUNC?`

5-56 :SENSe{1-16}:ABORtcal Subsystem

The :SENSe{1-16}:ABORtcal subsystem command aborts the calibration on the indicated channel.

Calibration Subsystems with Actual Calibrations

Related calibration subsystems that perform actual calibrations are:

- ":SENSe{1-16}:ABORtcal Subsystem" on page 5-264
- ":SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs" on page 5-279
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- ":SENSe{1-16}:CORRection:COLLect:FULL4 Subsystem - 4-Port VNAs" on page 5-289
- ":SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem" on page 5-317
- ":SENSe{1-16}:CORRection:COLLect:PORT Subsystem" on page 5-336
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380

:SENSe{1-16}:ABORtcal

Description: Aborts the current hardware or RF calibration on the indicated channel. No query.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS{1}:ABOR`

5-57 :SENSe{1-16}:AVERage Subsystem

The :SENSe{1-16}:AVERage subsystem commands configure and control the averaging function on the indicated channel.

Channel and Sweep Subsystems

Related channel and sweep configuration and control subsystems are:

- Section 5-57 :SENSe{1-16}:AVERage Subsystem on page 5-265
- Section 5-87 :SENSe{1-16}:FREQuency Subsystem on page 5-399
- Section 5-99 :SENSe{1-16}:SPUR Subsystem on page 5-489
- Section 5-100 :SENSe{1-16}:SWEep Subsystem on page 5-490

:SENSe{1-16}:AVERage:CLEar

Description: Clears and restarts the averaging sweep count of the given channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:AVER:CLE`

:SENSe{1-16}:AVERage:COUNt <NRf>
:SENSe{1-16}:AVERage:COUNt?

Description: Sets the averaging count for the indicated channel. The channel must be turned on. Outputs the averaging count for the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 1024

Default Value: 1

Syntax Example: `:SENS1:AVER:COUN 101`

`:SENS1:AVER:COUN?`

:SENSe{1-16}:AVERage:SWEep?

Description: Query only. Outputs the averaging sweep count for the given channel.

Query Parameters: <NR1> The output parameter is an integer.

Range: NA

Default Value: 0

Syntax Example: `:SENS1:AVER:SWE?`

:SENSe{1-16}:AVERage:TYPe <char>
:SENSe{1-16}:AVERage:TYPe?

Description: Sets the averaging function type to point-by-point or sweep-by-sweep. Outputs the averaging function type of point-by-point or sweep-by-sweep.

Cmd Parameters: <char> POINtbypoint | SWEepbysweep

Query Parameters: <char> POIN | SWE

Range: NA

Default Value: POIN

Syntax Example: `:SENS1:AVER:TYP POIN`

`:SENS1:AVER:TYP?`

:SENSe{1-16}:AVERage[:STATe] <char>
:SENSe{1-16}:AVERage[:STATe]?

Description: Turns averaging on/off for the given channel. Outputs the averaging function on/off status on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:AVER ON`

`:SENS1:AVER?`

5-58 :SENSe{1-16}:BANDwidth Subsystem

The :SENSe{1-16}:BANDwidth subsystem command sets the IF bandwidth. Note that this command is the same as the :SENSe{1-16}:BWIDth command:

IF Configuration Subsystems

Related IF configuration and control subsystems are:

- Section 5-54 :SENSe:BANDwidth Subsystem on page 5-260
- Section 5-58 :SENSe{1-16}:BANDwidth Subsystem on page 5-267
- Section 5-59 :SENSe{1-16}:BWIDth Subsystem on page 5-268

:SENSe{1-16}:BANDwidth[:RESolution] <NRf>
:SENSe{1-16}:BANDwidth[:RESolution]?

Description: The command sets the IF bandwidth for the given channel. The query outputs the IF bandwidth for the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 1 to 1E6. The system will automatically select the closest IF bandwidth from the following options:

- 1, 3, 10, 30, 100, 300 Hz
- 1, 3, 10, 30, 100, 300 kHz
- 1 MHz

Default Value: 1.00000000000E+003

Syntax Example: `:SENS1:BAND 4`

`:SENS1:BAND?`

Related Cmds: :SENSe{1-16}:BWIDth[:RESolution] <NRf>

5-59 :SENSe{1-16}:BWIDth Subsystem

The :SENSe{1-16}:BWIDth subsystem command sets the IF bandwidth. Note that this command is the same as the :SENSe{1-16}:BANDwidth command.

IF Configuration Subsystems

Related IF configuration and control subsystems are:

- Section 5-54 :SENSe:BANDwidth Subsystem on page 5-260
- Section 5-58 :SENSe{1-16}:BANDwidth Subsystem on page 5-267
- Section 5-59 :SENSe{1-16}:BWIDth Subsystem on page 5-268

:SENSe{1-16}:BWIDth[:RESolution] <NRf>
:SENSe{1-16}:BWIDth[:RESolution]?

Description: The command sets the IF bandwidth for the given channel. The query outputs the IF bandwidth for the given channel. A related command is:

- :SENSe{1-16}:BANDwidth[:RESolution] <NRf> on page 5-267

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 1 to 1E6. The system will automatically select the closest IF bandwidth from the following options:

- 1, 3, 10, 30, 100, 300 Hz
- 1, 3, 10, 30, 100, 300 kHz
- 1 MHz

Default Value: 1.00000000000E+003

Syntax Example: `:SENS1:BWID 3100`

`:SENS1:BWID?`

5-60 :SENSe{1-16}:CORRection:COEFficient:PORT Subsystem - Simulation

The :SENSe{1-16}:CORRection:COEFficient:PORT subsystem commands are used to simulate a instrument calibration on 2-Port or 4-Port VNA instrument so that a set of user-defined calibration parameters can be input using the :SENSe{1-16}:CORRection:COEFficient <char>,<block> command. Note that this subsystem does not perform an actual calibration.

Calibration Simulation Subsystems

These subsystems are used to create a calibrated state in the instrument which is followed by adding the required error correction coefficients for the required calibration type. If this approach is used, each error correction coefficient is entered by separate commands. Simulated calibration subsystems are:

- Section 5-13 :CALCulate{1-16}:FSIMulator:NETWork Subsystem - Simulation on page 5-51
- Section 5-14 :CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation on page 5-63
- Section 5-14 :CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation on page 5-63
- Section 5-60 :SENSe{1-16}:CORRection:COEFficient:PORT Subsystem - Simulation on page 5-269
- Section 5-61 :SENSe{1-16}:CORRection:COEFficient Subsystem - Simulation on page 5-274

Calibration Type Abbreviations

The calibration abbreviations and their calibration types are:

- :1P2PF refers to a one-path two-port calibration forward direction
- :1P2PR refers to a one path two port calibration reverse direction
- :FULL1 refers to a full one port calibration
- :FULL2 refers to a full two port calibration
- :FULL3 refers to a full three port calibration
- :FULLB refers to a full one port reflection calibration on both ports
- :FULL4 refers to a full four port calibration
- :RESP1 refers to a one port response calibration
- :RESPB refers to a one port response calibration both ports
- :TFRB refers to a transmission frequency response calibration both directions
- :TFRF refers to a transmission frequency response calibration forward direction
- :TFRR refers to a transmission frequency response calibration reverse direction

Each calibration simulation type command is described in greater detail in the individual command descriptions below.

Related Query

These commands set the Calibration Type. To query which Calibration Type has been set, use the command:

```
:SENSe{1-16}:CORRection:COLLect:TYPe?
```

:SENSe{1-16}:CORRection:COEFficient <char>, <block>
:SENSe{1-16}:CORRection:COEFficient? <char>

Description: The command inputs the values for a single calibration correction coefficient such as ED1. The command identifies the coefficient name and then the coefficient data for a calibration on the indicated channel. Separate commands are issued for each required correction coefficient. The query outputs the values of a correction coefficient of the calibration on the indicated channel.

- There are 12 correction coefficients available for a Full 2-Port Calibration which are: ED1, EP1S, ET11, ET21, EP2L, EX21, ED2, EP2S, ET22, ET12, EP1L, and EX12.
- There are 24 correction coefficients available for a Full 3-Port Calibration that include the 12 coefficients above and the 12 coefficients below: ED3, ET31, ET32, ET13, ET23, ET33, EP3L, EP3S, EX31, EX32, EX13, and EX23.
- There are 40 correction coefficients available for a Full 4-Port Calibration include the 24 coefficients above and the 16 coefficients below: ED4, ET14, ET41, ET24, ET42, ET34, ET43, ET44, EP4L, EP4S, EX14, EX24, EX34, EX41, EX42, and EX43.

Cmd Parameters: <char> ED1 | EP1S | ET11 | ET21 | EP2L | EX21 | ED2 | EP2S | ET22 | ET12 | EP1L | EX12 | ED3 | ET31 | ET32 | ET13 | ET23 | ET33 | EP3L | EP3S | EX31 | EX32 | EX13 | EX23 | ED4 | ET14 | ET41 | ET24 | ET42 | ET34 | ET43 | ET44 | EP4L | EP4S | EX14 | EX24 | EX34 | EX41 | EX42 | EX43

<block> The second input parameter is a block ASCII value. The actual ASCII block must be constructed and sent.

Query Parameters: <char> ED1 | EP1S | ET11 | ET21 | EP2L | EX21 | ED2 | EP2S | ET22 | ET12 | EP1L | EX12 | ED3 | ET31 | ET32 | ET13 | ET23 | ET33 | EP3L | EP3S | EX31 | EX32 | EX13 | EX23 | ED4 | ET14 | ET41 | ET24 | ET42 | ET34 | ET43 | ET44 | EP4L | EP4S | EX14 | EX24 | EX34 | EX41 | EX42 | EX43

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF ED1, <block>`

`:SENS1:CORR:COEF? ED1`

:SENSe{1-16}:CORRection:COEFficient:FULL4

Description: The command simulates a Full Four Port Calibration on the indicated Channel. This command requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:FULL4`

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PF

Description: The command simulates a One-Path Two-Port Calibration Forward Direction on the indicated port set on the indicated Channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PF`

or

`:SENSe{1-16}:CORRection:COLLect:CALB:1P2PF`

Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:PORT12:1P2PF`

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PR

Description: The command simulates a One-Path Two-Port Calibration Reverse Direction on the indicated port set on the indicated Channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:PORT12:1P2PR`

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2

Description: The command simulates a Full Two-Port Calibration on the indicated port set on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:PORT12:FULL2`

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULLB

Description: The command simulates a Full One-Port Reflection Calibration on both ports on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COEF:PORT12:FULLB

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:RESPB

Description: The command simulates a One-Port Response Calibration on both ports on the indicated port set on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

Parameters: NA

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COEF:PORT12:RESPB

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRB

Description: The command simulates a Transmission Frequency Response Calibration in both directions on the indicated port set on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

To query the state of this command use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COEF:PORT12:TFRB

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRF

Description: The command simulates a Transmission Frequency Response Calibration in the forward direction on the indicated port set on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

Parameters: NA

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COEF:PORT12:TFRF

:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRR

Description: The command simulates a Transmission Frequency Response Calibration in the reverse direction on the indicated port set on the indicated Channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:PORT12:TFRR`

:SENSe{1-16}:CORRection:COEFficient:PORT{123 | 124 | 134 | 234}:FULL3

Description: The command simulates a Full Three Port Calibration on the indicated port set on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:PORT123:FULL3`

:SENSe{1-16}:CORRection:COEFficient:PORT{1-4}:FULL1

Description: The command simulates a Full One-Port Reflection Calibration on the indicated port on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:PORT1:FULL1`

:SENSe{1-16}:CORRection:COEFficient:PORT{1-4}:RESP1

Description: The command simulates a One-Port Response Calibration on the indicated port on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COEF:PORT1:RESP1`

5-61 :SENSe{1-16}:CORRection:COEFficient Subsystem - Simulation

The :SENSe{1-16}:CORRection:COEFficient subsystem commands are used to simulate a instrument calibration on 2-Port VNA instruments so that a set of user-defined calibration parameters can be input using the :SENSe{1-16}:CORRection:COEFficient <char>,<block> command. Note that this subsystem does not perform an actual calibration.

Calibration Simulation Subsystems

These subsystems are used to create a calibrated state in the instrument which is followed by adding the required error correction coefficients for the required calibration type. If this approach is used, each error correction coefficient is entered by separate commands. Simulated calibration subsystems are:

- :CALCulate{1-16}:FSIMulator:NETWork Subsystem - Simulation on page 5-51
- :CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation on page 5-63
- :CALCulate{1-16}:FSIMulator:NETWork {1-50} Subsystem - Simulation on page 5-63
- :SENSe{1-16}:CORRection:COEFficient:PORT Subsystem - Simulation on page 5-269
- :SENSe{1-16}:CORRection:COEFficient Subsystem - Simulation on page 5-274

Calibration Type Abbreviations

The calibration abbreviations and their calibration types are:

- :1P2PF refers to a one-path two-port calibration forward direction
- :1P2PR refers to a one path two port calibration reverse direction
- :FULL1 refers to a full one port calibration
- :FULL2 refers to a full two port calibration
- :FULL3 refers to a full three port calibration
- :FULLB refers to a full one port reflection calibration on both ports
- :FULL4 refers to a full four port calibration
- :RESP1 refers to a one port response calibration
- :RESPB refers to a one port response calibration both ports
- :TFRB refers to a transmission frequency response calibration both directions
- :TFRF refers to a transmission frequency response calibration forward direction
- :TFRR refers to a transmission frequency response calibration reverse direction

Each calibration simulation type command is described in greater detail in the individual command descriptions below.

These commands set the Calibration Type. To query which Calibration Type has been sect, use the command:

```
:SENSe{1-16}:CORRection:COLLect:TYPe?
```

:SENSe{1-16}:CORRection:COEFficient:1P2PF

Description: The command simulates a one-path two-port calibration forward direction on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port VNA or a 4-Port VNA instrument. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:1P2PF`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:1P2PF`

:SENSe{1-16}:CORRection:COEFficient:1P2PR

Description: The command simulates a one-path Two-Port Calibration Reverse Direction on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:1P2PR`

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:1P2PR`

:SENSe{1-16}:CORRection:COEFficient:FULL1

Description: The command simulates a full One-Port Reflection Calibration on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port VNA or a 4-Port VNA instrument. No query. Before sending this command, the simulation port must be specified using:

`:SENSe{1-16}:CORRection:COLLect:PORT`

To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:FULL1`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:FULL1`

:SENSe{1-16}:CORRection:COEFficient:FULL2

Description: The command simulates a full Two-Port Calibration on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:FULL2`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:FULL2`

:SENSe{1-16}:CORRection:COEFficient:FULLB

Description: The command simulates a full One-Port Reflection Calibration on both ports on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:FULLB`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:FULLB`

:SENSe{1-16}:CORRection:COEFficient:RESP1

Description: The command simulates a One-Port Response Calibration on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query.

Before sending this command, the simulation port must be specified using:

`:SENSe{1-16}:CORRection:COLLect:PORT`

To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:RESP1`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:RESP1`

:SENSe{1-16}:CORRection:COEFficient:RESPB

Description: The command simulates a One-Port Response Calibration on both ports on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:RESPB`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:RESPB`

:SENSe{1-16}:CORRection:COEFficient:TFRB

Description: The command simulates a Transmission Frequency Response Calibration in both directions on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:TFRB`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:TFRB`

:SENSe{1-16}:CORRection:COEFficient:TFRF

Description: The command simulates a Transmission Frequency Response Calibration in the forward direction on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To set this calibration mode and then perform an actual calibration, use:

`:SENSe{1-16}:CORRection:COLLect:TFRF`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:TFRF`

:SENSe{1-16}:CORRection:COEFficient:TFRR

Description: The command simulates a Transmission Frequency Response Calibration in the reverse direction on the indicated channel. The port pair is limited to Ports 1-2 but can be used on either a 2-Port or 4-Port VNA instrument. No query. To query the state of this command, use:

```
:SENSe{1-16}:CORRection:COLLect:TYPe?
```

To set this calibration mode and then perform an actual calibration, use:

```
:SENSe{1-16}:CORRection:COLLect:TFRR
```

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COEF:TFRR`

5-62 :SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs

The :SENSe{1-16}:CORRection:COLLect:CALB subsystem commands set the calibration type for a second calibration (termed :CALB) and can only be used on a 4-Port VNA instrument. Once the command is issued, the calibration proceeds.

Calibration Subsystems with Actual Calibration

Related calibration subsystems that perform actual calibrations are:

- :SENSe{1-16}:ABORtcal Subsystem on page 5-264
- :SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs on page 5-279
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- :SENSe{1-16}:CORRection:COLLect:FULL4 Subsystem - 4-Port VNAs on page 5-289
- :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem on page 5-317
- :SENSe{1-16}:CORRection:COLLect:PORT Subsystem on page 5-336
- :SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem on page 5-380

Calibration Type Abbreviations

The calibration abbreviations and their calibration types are:

- :1P2PF refers to a one-path two-port calibration forward direction
- :1P2PR refers to a one path two port calibration reverse direction
- :FULL2 refers to a full two port calibration

Related Query

Most calibration commands of this type do not have a directly related query. To query the state of these commands, use: `:SENSe{1-16}:CORRection:COLLect:TYPe?`

:SENSe{1-16}:CORRection:COLLect:CALB:1P2PF

Description: This command requires a 4-Port VNA instrument. The command sets the second 2-Port calibration to One-Path Two-Port Forward on the indicated channel and port pair. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:CALB:1P2PF`

:SENSe{1-16}:CORRection:COLLect:CALB:1P2PR

Description: This command requires a 4-Port VNA instrument. The command sets the second 2-Port calibration to One-Path Two-Port Reverse on the indicated channel and port pair. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:CALB:1P2PR`

:SENSe{1-16}:CORRection:COLLect:CALB:FULL2

Description: This command requires a 4-Port VNA instrument. The command sets the second 2-Port calibration to Full Two-Port on the indicated channel and port pair. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:CALB:FULL2`

5-63 :SENSe{1-16}:CORRection:COLLect:ECAL Subsystem - 4-Port VNAs

The :SENSe{1-16}:CORRection:COLLect:ECAL subsystem commands set the automatic calibration parameters.

:SENSe{1-16}:CORRection:COLLect:ECAL:AUTOmatic:ORIentation[:STATe] <char>

:SENSe{1-16}:CORRection:COLLect:ECAL:AUTOmatic:ORIentation[:STATe]?

Description: The command turns automatic Autocal module orientation detection on/off for the given channel. Query outputs the on/off status of the Autocal module orientation detection on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default: NA

Syntax Example: SENS1:CORR:COLL:ECAL:AUTO:ORI ON

SENS1:CORR:COLL:ECAL:AUTO:ORI?

:SENSe{1-16}:CORRection:COLLect:ECAL:BEGin?

Description: The query starts an Autocal calibration and returns status or next step information.

Cmd Parameters: NA

Query Parameters: <NR1> Outputs one of the following ECAL List of Messages:

0 - AssurancePassed

1 - Update

2 - TrueThru

3 - Adapter

4 - NoModule

5 - NoOrient

6 - NoFile

7 - NoMatch

8 - No12T

9 - NotAllowed

10 - OutOfRange

11 - AssuranceFailed

12 - Aborted

13 - AbortOK

14 - AbortNotOK

15 - ACError

16 - ACFatalError

17 - DoneCalculateCoeff

18 - ACConnectCalB

19 - CharacBad

20 - DisplayMessage

21 - ConnectToPort1

22 - ConnectToPort2

23 - ConnectToPort3

24 - ConnectToPort4

25 - ConnectToPorts12

26 - ConnectToPorts13

27 - ConnectToPorts14

28 - ConnectToPorts23

29 - ConnectToPorts24

30 - ConnectToPorts34

31 - ConnectThrubwPorts12

32 - ConnectThrubwPorts13

33 - ConnectThrubwPorts14

34 - ConnectThrubwPorts23

35 - ConnectThrubwPorts24

36 - ConnectThrubwPorts34

37 - SequentialBegins

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COLL:ECAL:BEG?

:SENSe{1-16}:CORRection:COLLect:ECAL:CALB:TRUEthru <char>

Description: The command turns on/off the use of TrueThrus during Autocal CALB Calibrations on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COLL:ECAL:CALB:TRUE ON

:SENSe{1-16}:CORRection:COLLect:ECAL:CHARacterize?

Description: The query starts an Autocal characterization and returns status or next step information.

Cmd Parameters: NA

Query Parameters: <NR1> Outputs one of the following ECAL List of Messages:

0 - AssurancePassed

1 - Update

2 - TrueThru

3 - Adapter

4 - NoModule

5 - NoOrient

6 - NoFile

7 - NoMatch

8 - No12T

9 - NotAllowed

10 - OutOfRange

11 - AssuranceFailed

12 - Aborted

13 - AbortOK

14 - AbortNotOK

15 - ACError

16 - ACFatalError

17 - DoneCalculateCoeff

18 - ACConnectCalB

19 - CharacBad

20 - DisplayMessage

21 - ConnectToPort1

22 - ConnectToPort2

23 - ConnectToPort3

24 - ConnectToPort4

25 - ConnectToPorts12

26 - ConnectToPorts13

27 - ConnectToPorts14

28 - ConnectToPorts23

29 - ConnectToPorts24

30 - ConnectToPorts34

31 - ConnectThrubwPorts12

32 - ConnectThrubwPorts13

33 - ConnectThrubwPorts14

34 - ConnectThrubwPorts23

35 - ConnectThrubwPorts24

36 - ConnectThrubwPorts34

37 - SequentialBegins

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COLL:ECAL:CHAR?

:SENSe{1-16}:CORRection:COLLect:ECAL:CONTinue?

Description: The query starts an Autocal characterization and returns status or next step information.

Cmd Parameters: NA

Query Parameters: <NR1> Outputs one of the following ECAL List of Messages:

0 - AssurancePassed

1 - Update

2 - TrueThru

3 - Adapter

4 - NoModule

5 - NoOrient

6 - NoFile

7 - NoMatch

8 - No12T

9 - NotAllowed

10 - OutOfRange

11 - AssuranceFailed

12 - Aborted

13 - AbortOK

14 - AbortNotOK

15 - ACError

16 - ACFatalError

17 - DoneCalculateCoeff

18 - ACConnectCalB

19 - CharacBad

20 - DisplayMessage

21 - ConnectToPort1

22 - ConnectToPort2

23 - ConnectToPort3

24 - ConnectToPort4

25 - ConnectToPorts12

26 - ConnectToPorts13

27 - ConnectToPorts14

28 - ConnectToPorts23

29 - ConnectToPorts24

30 - ConnectToPorts34

31 - ConnectThrubwPorts12

32 - ConnectThrubwPorts13

33 - ConnectThrubwPorts14

34 - ConnectThrubwPorts23

35 - ConnectThrubwPorts24

36 - ConnectThrubwPorts34

37 - SequentialBegins

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COLL:ECAL:CONT?

:SENSe{1-16}:CORRection:COLLect:ECAL:MSGS:LIST?

Description: The query outputs a copy of the Autocal messages list

Cmd Parameters: NA

Query Parameters: <NR1> Outputs ECAL List of Messages:

0 - AssurancePassed

1 - Update

2 - TrueThru

3 - Adapter

4 - NoModule

5 - NoOrient

6 - NoFile

7 - NoMatch

8 - No12T

9 - NotAllowed

10 - OutOfRange

11 - AssuranceFailed

12 - Aborted

13 - AbortOK

14 - AbortNotOK

15 - ACError

16 - ACFatalError

17 - DoneCalculateCoeff

18 - ACConnectCalB

19 - CharacBad

20 - DisplayMessage

21 - ConnectToPort1

22 - ConnectToPort2

23 - ConnectToPort3

24 - ConnectToPort4

25 - ConnectToPorts12

26 - ConnectToPorts13

27 - ConnectToPorts14

28 - ConnectToPorts23

29 - ConnectToPorts24

30 - ConnectToPorts34

31 - ConnectThrubwPorts12

32 - ConnectThrubwPorts13

33 - ConnectThrubwPorts14

34 - ConnectThrubwPorts23

35 - ConnectThrubwPorts24

36 - ConnectThrubwPorts34

37 - SequentialBegins

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COLL:ECAL:MSGS:LIST?

:SENSe{1-16}:CORRection:COLLect:ECAL:ORIentation <char>
:SENSe{1-16}:CORRection:COLLect:ECAL:ORIentation?

Description: The command turns Autocal module orientation detection off and sets the orientations manually for the given channel. A query outputs the Autocal module orientations of the given channel.

Cmd Parameters: <char> L1 | L2 | L3 | L4 | R1 | R2 | R3 | R4 | L1R2 | L1R3 | L1R4 | L2R3 | L2R4 | L3R4 | R1L2 | R1L3 | R1L4 | R2L3 | R2L4 | R3L4 | R2L1 | R3L1 | R4L1 | R3L2 | R4L2 | R4L3 | L2R1 | L3R1 | L4R1 | L3R2 | L4R2 | L4R3

Query Parameters: <char> L1 | L2 | L3 | L4 | R1 | R2 | R3 | R4 | L1R2 | L1R3 | L1R4 | L2R3 | L2R4 | L3R4 | R1L2 | R1L3 | R1L4 | R2L3 | R2L4 | R3L4 | R2L1 | R3L1 | R4L1 | R3L2 | R4L2 | R4L3 | L2R1 | L3R1 | L4R1 | L3R2 | L4R2 | L4R3

Range: NA

Default: L1R2

Syntax Example: :SENS1:CORR:COLL:ECAL:ORI L2R4

:SENS1:CORR:COLL:ECAL:ORI?

:SENSe{1-16}:CORRection:COLLect:ECAL:PORT{12|13|14|23|24|34}:FULL4

Description: The command sets FULL4 calibration type with the given CALa portset and indicated channel.

Cmd Parameters: <char> 12|13|14|23|24|34

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COLL:ECAL:PORT12:FULL4

:SENSe{1-16}:CORRection:COLLect:ECAL[:CALa]:TRUEthru <char>
:SENSe{1-16}:CORRection:COLLect:ECAL[:CALa]:TRUEthru?

Description: The command turns on/off the use of TrueThru during Autocal CALa Calibration on the given channel.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default: 0

Syntax Example: :SENS1:CORR:COLL:ECAL:TRUE ON

:SENS1:CORR:COLL:ECAL:TRUE?

5-64 :SENSe{1-16}:CORRection:COLLect:DISPlay Subsystem

The :SENSe{1-16}:CORRection:COLLect:DISPLay subsystem controls the front panel display on the given channel.

:SENSe{1-16}:CORRection:COLLect:DISPlay[:CURRent]:CALibration[:STATe] <char>
:SENSe{1-16}:CORRection:COLLect:DISPlay[:CURRent]:CALibration[:STATe]?

Description: Turns on/off the state of displaying the current calibration while performing a new calibration on the given channel. Outputs the on/off state of displaying the current calibration while performing a new calibration on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:CORR:COLL:DISP:CAL 1`

`:SENS1:CORR:COLL:DISP:CAL?`

5-65 :SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem

The :SENSe{1-16}:CORRection:COLLect:FLEXible command defines the S-Parameter set for the existing calibration type on the given channel.

Calibration Option Subsystems

Related calibration option configuration and control subsystems are:

- :CALCulate{1-16}:CORRection Subsystem - Adapters/Merge Calibration on page 5-14
- :CALCulate{1-16}:EXTRaction Subsystem - Network Extraction on page 5-33
- :CALCulate{1-16}:NXN Subsystem on page 5-88
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- :SENSe{1-16}:CORRection:COLLect:HYBRid Subsystem on page 5-290

:SENSe{1-16}:CORRection:COLLect:FLEXible:DEFine <char>{,<char2>,...,<charn>}
:SENSe{1-16}:CORRection:COLLect:FLEXible:DEFine?

Description: The command defines the S-Parameter set for the existing calibration type on the given channel. At least one S-Parameter must be selected. Up to 16 S-Parameters may be selected in any sequence. The use of S-Parameters with Port 3 and/or Port 4 require a 4-Port VNA instrument. The query outputs the S-Parameter set defined for the existing calibration type on the given channel.

Cmd Parameters: <char>{,<char2>,...<char6>} S11 | S12 | S21 | S22 | S13 | S31 | S23 | S32 | S33 | S14 | S41 | S24 | S42 | S34 | S43 | S44

Query Parameters: <char>{,<char2>,...<char6>} S11 | S12 | S21 | S22 | S13 | S31 | S23 | S32 | S33 | S14 | S41 | S24 | S42 | S34 | S43 | S44

Range: NA

Default: S11

Syntax Example: `:SENS1:CORR:COLL:FLEX:DEF S11, S12`

`:SENS1:CORR:COLL:FLEX:DEF?`

:SENSe{1-16}:CORRection:COLLect:FLEXible[:STATe] <char>
:SENSe{1-16}:CORRection:COLLect:FLEXible[:STATe]?

Description: The command toggles on/off the flexible definition of a calibration on the given channel. The query outputs the on/off status of the flexible definition of a calibration on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: `:SENS1:CORR:COLL:FLEX ON`

`:SENS1:CORR:COLL:FLEX?`

5-66 :SENSe{1-16}:CORRection:COLLect:FULL4 Subsystem - 4-Port VNAs

The :SENSe{1-16}:CORRection:COLLect:FULL4 sets the calibration type to Full 4-Port Calibration for the indicated channel. The command requires a 4-Port VNA instrument.

Calibration Subsystems with Actual Calibration

Related calibration subsystems that perform actual calibrations are:

- :SENSe{1-16}:ABORtcal Subsystem on page 5-264
- :SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs on page 5-279
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- :SENSe{1-16}:CORRection:COLLect:FULL4 Subsystem - 4-Port VNAs on page 5-289
- :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem on page 5-317
- :SENSe{1-16}:CORRection:COLLect:PORT Subsystem on page 5-336
- :SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem on page 5-380

:SENSe{1-16}:CORRection:COLLect:FULL4

Description: The command sets the calibration type to Full Four Port for the indicated channel. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:FULL4`

5-67 :SENSe{1-16}:CORRection:COLLect:HYBRid Subsystem

The :SENSe{1-16}:CORRection:COLLect:HYBRid subsystem commands configure, control, and execute hybrid calibration on 2-Port or 4-Port VNA instruments.

Calibration Option Subsystems

Related calibration option configuration and control subsystems are:

- :CALCulate{1-16}:CORRection Subsystem - Adapters/Merge Calibration on page 5-14
- :CALCulate{1-16}:EXTRaction Subsystem - Network Extraction on page 5-33
- :CALCulate{1-16}:NXN Subsystem on page 5-88
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- :SENSe{1-16}:CORRection:COLLect:HYBRid Subsystem on page 5-290

Calibration Type Abbreviations

The calibration abbreviations and their calibration types are:

- :FULL2 refers to a full two port calibration
- :FULL3 refers to a full three port calibration
- :FULL4 refers to a full four port calibration

:SENSe{1-16}:CORRection:COLLect:HYBRid:FILe{1-4} <string>
:SENSe{1-16}:CORRection:COLLect:HYBRid:FILe{1-4}?

Description: Sets the file path of the indicated calibration file needed to hybridize for the indicated port or port-pair on the indicated channel. Outputs the file path of the calibration file needed to hybridize for the indicated port or port-pair on the indicated channel.

Cmd Parameters: <string> Filename and path in the form 'x:\directory\filename.xxx' where x:\directory\ must exist. See definition of "<string>" on page 2-12.

Query Parameters: <string> Filename and path in the form 'x:\directory\filename.xxx'.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:HYBR:FIL1 'C:\directory\filename.xxx'`

`:SENS1:CORR:COLL:HYBR:FIL1?`

:SENSe{1-16}:CORRection:COLLect:HYBRid:FULL4

Description: Begins a Hybrid FULL4 port calibration using four FULL1 calibrations on all ports and the channel indicated. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:HYBR:FULL4`

:SENSe{1-16}:CORRection:COLLect:HYBRid:MULTiple:THRu <char> {,<char>,...,<char>}
:SENSe{1-16}:CORRection:COLLect:HYBRid:MULTiple:THRu?

Description: The command adds one or more Transmission Throughs (Thrus) to the Hybrid Calibration process on the indicated channel where:

- THRu12 or THR12 = Sets the through line between Port 1 and Port 2.
- THRu13 or THR13 = Sets the through line between Port 1 and Port 3.
- THRu14 or THR14 = Sets the through line between Port 1 and Port 4.
- THRu23 or THR23 = Sets the through line between Port 2 and Port 3.
- THRu24 or THR24 = Sets the through line between Port 2 and Port 4.
- THRu34 or THR34 = Sets the through line between Port 3 and Port 4.

The use of THRus connected to Port 3 and/or Port 4 require a 4-Port VNA instrument. At least one THRu must be defined. Up to 6 THRus can be defined. The query outputs the Hybrid calibration process list of Transmission Thrus on the indicated channel.

Cmd Parameters: <char> THRu12 | THRu13 | THRu14 | THRu23 | THRu24 | THRu34

Query Parameters: <char> THR12 | THR13 | THR14 | THR23 | THR24 | THR34

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:HYBR:MULT:THR THR12, THR13, THR14`

`:SENS1:CORR:COLL:HYBR:MULT:THR?`

:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2

Description: Begins a Hybrid FULL2 calibration using two FULL1 calibrations on the indicated port pair and channel. No query. To query the state of this command use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

There are six possible FULL2 port-pair combinations available on a 4-Port VNA instrument:

- 12 = Port Pair 1-2
- 13 = Port Pair 1-3
- 14 = Port Pair 1-4
- 23 = Port Pair 2-3
- 24 = Port Pair 2-4
- 34 = Port Pair 3-4.

Prepare for this command by specifying the Full2 calibration file name for each port. For example, if Port Pair 2-4 is to be calibrated on channel 3, issue the two commands as:

`:SENSe{3}:CORRection:COLLect:HYBRid:FILe1‘C:\filename1.xxx’`

`:SENSe{3}:CORRection:COLLect:HYBRid:FILe2‘C:\filename2.xxx’`

Use filename1 for the first number in the pair and use filename2 for the second number in the pair.

No query.

Cmd Parameters: <char> 12 | 13 | 14 | 23 | 24 | 34

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: :SENS1:CORR:COLL:HYBR:PORT12:FULL2

:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{123 | 124 | 134 | 234}:FULL3

Description: Begins a Hybrid FULL3 calibration using three FULL1 calibrations on the triport and channel indicated. No query. To query the state of this command use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

There are four possible FULL3 port triplet variations available on a Four-Port Test Set:

- 123 = Port Triplet 1-2-3
- 124 = Port Triplet 1-2-4
- 134 = Port Triplet 1-3-4
- 234 = Port Triplet 2-3-4.

Prepare for this command by specifying the Full1 (Full One) calibration file name for each port. For example, if port triplet 123 is to be calibrated on Channel 4, issue the three commands as:

:SENSe{4}:CORRection:COLLect:HYBRid:FILe1'C:\filename1.xxx'

:SENSe{4}:CORRection:COLLect:HYBRid:FILe2'C:\filename2.xxx'

:SENSe{4}:CORRection:COLLect:HYBRid:FILe3'C:\filename3.xxx'

Use FILe1 for the first number in the port triplet set, use FILe2 for the second number in the port triplet set, and FILe3 for the third number in the port triplet set.

Cmd Parameters: <char> 123 | 124 | 134 | 234

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: :SENS1:CORR:COLL:HYBR:PORT123:FULL3

:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL4

Description: Begins a Hybrid FULL4 calibration using two FULL2 calibrations on the port pair and channel indicated. No query. To query the state of this command use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

There are six possible FULL2 port pair combinations available on a Four-Port Test Set:

- 12 = Port Pair 1-2
- 13 = Port Pair 1-3
- 14 = Port Pair 1-4
- 23 = Port Pair 2-3
- 24 = Port Pair 24
- 34 = Port Pair 3-4

Prepare for this command by specifying the Full2 calibration file name for each port. For example, if Ports 1-3 and Ports 2-4 are to be calibrated on Channel 1, issue the two commands as:

:SENSe{1}:CORRection:COLLect:HYBRid:FILe1‘C:\filename1.xxx’

:SENSe{1}:CORRection:COLLect:HYBRid:FILe2‘C:\filename2.xxx’

Use FILe1 for the first port pair Port 1-3 and use FILe2 for the second port pair Port2-4.

Cmd Parameters: <char> 12 | 13 | 14 | 23 | 24 | 34

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: :SENS1:CORR:COLL:HYBR:PORT12:FULL4

:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu

Description: Initiates collection of the through standard data for the Hybrid Calibration on the indicated port pair and channel. No query. There are six possible port pair combinations:

- 12 = Port Pair 1-2
- 13 = Port Pair 1-3
- 14 = Port Pair 1-4
- 23 = Port Pair 2-3
- 24 = Port Pair 24
- 34 = Port Pair 3-4

Cmd Parameters: <char> 12 | 13 | 14 | 23 | 24 | 34

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: :SENS1:CORR:COLL:HYBR:PORT12:THR

:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal[:STATe] <char>

:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal[:STATe]?

Description: The command sets the thru-line use reciprocal flag on the selected port-pair of the indicated channel.

The Query outputs the thru-line use reciprocal flag on the selected port-pair of the indicated channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Range: NA

Default Value: 0

Syntax Example: :SENS1:CORR:COLL:HYBR:PORT12:THR:RECIP ON

:SENS1:CORR:COLL:HYBR:PORT12:THR:RECIP?

5-68 :SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem

The :SENSe{1-16}:CORRection:COLLect:LRL:CALB subsystem commands provide control of Line-Reflect-Line second calibration (or CALB) configuration parameters, execution, and output reporting. The :LRL:CALB commands require a 4-Port VNA instrument.

LRL Calibration Subsystems

The LRL-related calibration commands are organized into five (5) subsystems in the following sections:

- Section 5-68 :SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem on page 5-295 (this subsystem)
- Section 5-69 :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem on page 5-305
- Section 5-70 :SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs on page 5-309
- Section 5-71 :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs on page 5-311
- Section 5-73 :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem on page 5-317

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND:COUNt <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND:COUNt?

Description: This command requires a 4-Port VNA instrument. The command sets the number of bands to use in the LRL CALB calibration on the given channel. The query outputs the number of bands to use in the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is an integer.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 5

Default: 1

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:BAND:COUN 2`

`:SENS1:CORR:COLL:LRL:CALB:BAND:COUN?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND{1-5}:REFLection:TYPe <char>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND{1-5}:REFLection:TYPe?

Description: This command requires a 4-Port VNA instrument. The command sets the Band reflection type of the LRL CALB calibration on the given channel where the type can be set as:

- OPENlike = The reflection type is more like an Open.
- SHORTlike = The reflection tyoe is more like a Short.
- BOTH = The reflection type is like both an Open and a Short.

The query outputs the reflection type of the LRL CALB calibration on the given channel for a given band.

Cmd Parameters: <char> OPENlike | SHORTlike | BOTH

Query Output: <char> OPEN | SHORT | BOTH

Range: NA

Default: OPEN

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:BAND1:REFL:TYP OPENlike`

`:SENS1:CORR:COLL:LRL:CALB:BAND1:REFL:TYP?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:LOAD <string>

Description: This command loads an LRL cal kit file into CALB LRL calibration on the given channel.

Cmd Parameters: <string> Filename and path to the file in the form:

`x:\directory\filename.lcf`

where `x:\directory\filename.lcf` exist and `filename.lcf` is a valid LRL cal kit file.

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:CKIT:LOAD c:\filename.lcf`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:NAMe <char>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:NAMe?

Description: This command sets the name that will be stored in the LRL cal kit file.

The query outputs the name that was last set for the currently loaded file.

Cmd Parameters: <char> A name that will be associated with the current cal kit file.

Query Output: <char> The name that is currently associated with the current cal kit file.

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:CKIT:NAM mylrlcalkit`

`:SENS1:CORR:COLL:LRL:CKIT:NAM?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:SAVe <string>

Description: This command saves an LRL cal kit file in the CALB LRL calibration on the given channel.

Cmd Parameters: <string> Filename and path to the file in the form:

`x:\directory\filename.lcf`

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:CKIT:SAV c:\filename.lcf`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE

Description: This command requires a 4-Port VNA instrument. The command initiates collection of the indicated Device Line Standard data for the CALB calibration on the indicated channel. No query.

Cmd Parameters: NA

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:LINE`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:FREQuency <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:FREQuency?

Description: This command requires a 4-Port VNA instrument. The command sets the reference frequency for loss of the given device of the LRL CALB calibration on the given channel. The query outputs the reference frequency for loss of the given device of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:LINE:FREQ 1.0E7`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:LINE:FREQ?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LENGth <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LENGth?

Description: This command requires a 4-Port VNA instrument. The command sets the line length of the given device of the LRL CALB calibration on the given channel. The query outputs the line length of the given device of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:LINE:LENG 2.5E-2`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:LINE:LENG?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LOSS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LOSS?

Description: This command requires a 4-Port VNA instrument. The command sets the line loss of the given device of the LRL CALB calibration on the given channel. The query outputs the line loss of the given device of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: <NR3> The output parameter is in dB/mm.

Range: MPND/1000 (MPND divided by 1000)

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:LINE:LOSS 3.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:LINE:LOSS?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C0?

Description: This command requires a 4-Port VNA instrument. The command sets the capacitance of the match device on the given port of the LRL CALB calibration on the given channel. The query outputs the capacitance of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: <NR3> The output parameter is in Farads.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C0 3.01E-12`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C0?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C1?

Description: This command requires a 4-Port VNA instrument. Sets the C1 coefficient of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the C1 coefficient of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz.

Query Parameters: <NR3> The output parameter is in Farads/Hertz.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C1 2.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C1?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C2?

Description: This command requires a 4-Port VNA instrument. Sets the C2 coefficient of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the C2 coefficient of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^2.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^2.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C2 2.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C2?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C3?

Description: This command requires a 4-Port VNA instrument. Sets the C3 coefficient of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the C3 coefficient of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^3.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^3.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C3 2.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:C3?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L0?

Description: This command requires a 4-Port VNA instrument. The command sets the inductance of the match device on the given port of the LRL CALB calibration on the given channel. Use of Port 3 or Port 4 requires a four-port instrument. The query outputs inductance of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L0?`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L0 2.0E-6`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L1?

Description: This command requires a 4-Port VNA instrument. Sets the L1 coefficient of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the L1 coefficient of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L1 1.4`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L1?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L2?

Description: This command requires a 4-Port VNA instrument. Sets the L2 coefficient of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the L2 coefficient of the match device on the given port of the LRL CALB calibration on the given channel

Cmd Parameters: <NRf> in units of Meters/Hertz^2 (Meters per Hertz squared)

Query Parameters: <NR3> in units of Meters/Hertz^2

Range: MPND with units of Meters/Hertz^2

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L2 10.3E-10`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L2?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L3?

Description: This command requires a 4-Port VNA instrument. Sets the L3 coefficient of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the L3 coefficient of the match device on the given port of the LRL CALB calibration on the given channel

Cmd Parameters: <NRf> in Henrys/Hertz^3 (Meters per Hertz cubed)

Query Parameters: <NR3> in Henrys/Hertz^3

Range: MPND

Default Value: 0

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L3 1.32E-25`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:L3?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF1?

Description: This command requires a 4-Port VNA instrument. Sets the offset length coefficient1 of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the offset length coefficient1 of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFF1 1.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFF1?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF2?

Description: This command requires a 4-Port VNA instrument. Sets the offset length coefficient2 of the match device on the given port of the LRL CALB calibration on the given channel.Outputs the offset length coefficient2 of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFF2 1.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFF2?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF3?

Description: This command requires a 4-Port VNA instrument. Sets the offset length coefficient3 of the match device on the given port of the LRL CALB calibration on the given channel. Outputs the offset length coefficient3 of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFF3 1.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFF3?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFFS?

Description: This command requires a 4-Port VNA instrument. The command sets the offset length of the match device on the given port of the LRL CALB calibration on the given channel. The query outputs the offset length of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFFS 1.0E0`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:OFFS?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:R <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:R?

Description: This command requires a 4-Port VNA instrument. The command sets the resistance of the match device on the given port of the LRL CALB calibration on the given channel. The query outputs resistance of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:R 7.5E1`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:R?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:Z0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:Z0?

Description: This command requires a 4-Port VNA instrument. The command sets the impedance of the match device on the given port of the LRL CALB calibration on the given channel. The query outputs the impedance of the match device on the given port of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:Z0 7.5E1`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:PORT1:MATCH:Z0?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:TYPe <char>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:TYPe?

Description: This command requires a 4-Port VNA instrument. The command sets the type of the given device in the LRL CALB calibration on the given channel. The query outputs the type of the given device in the LRL CALB calibration on the given channel.

Cmd Parameters: <char> LINE | MATCH | DEVICE1 | DEVICE2

Query Parameters: <char> LINE | MATCH | DEVICE1 | DEVICE2

Range: NA

Default: LINE

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:DEV1:TYP LINE`

`:SENS1:CORR:COLL:LRL:CALB:DEV1:TYP?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint?

Description: This command requires a 4-Port VNA instrument. The command sets the breakpoint frequency of the LRL CALB calibration on the given channel. The query outputs the breakpoint frequency of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default: 3.000000000E+009

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:FREQ:BRE 1.0E7`

`:SENS1:CORR:COLL:LRL:CALB:FREQ:BRE?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint{1-4} <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint{1-4}?

Description: This command sets the breakpoint frequency between bands of LRL CALB calibration on the given channel.

The query outputs the breakpoint frequency between bands of LRL CALB calibration on the given channel.

In the command, the number next to BREakpoint has the following correspondence:

- BREakpoint1: Breakpoint frequency between bands 2 and 1
- BREakpoint2: Breakpoint frequency between bands 3 and 2
- BREakpoint3: Breakpoint frequency between bands 4 and 3
- BREakpoint1: Breakpoint frequency between bands 5 and 4

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default: 3E09

Syntax Example: `:SENS:CORR:COLL:LRL:CALB:FREQ:BRE1 4E09`

`:SENS:CORR:COLL:LRL:CALB:FREQ:BRE1?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:OPEN:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:OPEN:OFFS?

Description: This command requires a 4-Port VNA instrument. The command sets the offset length of the open like reflection of the LRL CALB calibration on the given channel. The query outputs the offset length of the open like reflection of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:OPEN:OFFS 1.0E0`

`:SENS1:CORR:COLL:LRL:CALB:OPEN:OFFS?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:REFPlane <char>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:REFPlane?

Description: This command requires a 4-Port VNA instrument. The command sets the location of the reference plane in the LRL CALB calibration on the given channel to either the middle or the end of the calibration line. The query outputs the location of the reference plane in the LRL CALB calibration on the given channel.

Cmd Parameters: <char> MIDdle | END

Query Parameters: <char> MID | END

Range: NA

Default: END

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:REFP MID`

`:SENS1:CORR:COLL:LRL:CALB:REFP?`

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:SHORT:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:SHORT:OFFS?

Description: This command requires a 4-Port VNA instrument. The command sets the offset length of the short like reflection of the LRL CALB calibration on the given channel. The query outputs the offset length of the short like reflection of the LRL CALB calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:CALB:SHORT:OFFS 1.0E0`

`:SENS1:CORR:COLL:LRL:CALB:SHORT:OFFS?`

5-69 :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem

The :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} subsystem commands provide control of Line-Reflect-Line device parameters, configuration, execution, and output reporting. Some commands require a 4-Port VNA instrument. *Note: Devices 5 to 10 only available on 2 Port systems.*

LRL Calibration Subsystems

The LRL-related calibration commands are organized into five (5) subsystems in the following sections:

- Section 5-68 :SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem on page 5-295
- Section 5-69 :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem on page 5-305
- Section 5-70 :SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs on page 5-309
- Section 5-71 :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs on page 5-311
- Section 5-73 :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem on page 5-317

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:MATCH:PORT <char>
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:MATCH:PORT?

Description: The command sets the match-port to measure for a match device with both reflection types in the LRL calibration on the given channel. The available ports are Port 1 or Port 2. The command can be used by 2-Port and 4-Port VNA instruments.

The query outputs the match-port to measure for a match device with both reflection types in the LRL calibration on the given channel.

Cmd Parameters: <char> PORT1 | PORT2

Query Parameters: <char> PORT1 | PORT2

Range: NA

Default Value: PORT1

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:MATCH:PORT PORT1`

`:SENS1:CORR:COLL:LRL:DEV1:MATCH:PORT?`

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT{1-4}:MATCH

Description: The command initiates collection of the indicated device match standard data for the LRL calibration on the given channel and port. The use of Port 3 or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH`

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE

Description: The command initiates collection of the indicated device line standard data for the LRL calibration on the given channel and port-pair. The available port pair is 12 where it is fixed as Port 1 and Port 2. The command can be used by 2-Port and 4-Port VNA instruments. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT12:LINE`

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:DELay <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:DELay?

Description: The command sets the line delay of the given device for LRL calibration on the given channel.

The query outputs the line delay of the given device of the LRL calibration on the given channel

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: MPND

Default Value: Default value is dependent on the default dielectric of the substrate and the effective length default value.

Syntax Example: `:SENSe:CORR:COLL:LRL:DEV3:PORT12:LINE:DEL 20E-3`

`:SENSe:CORR:COLL:LRL:DEV3:PORT12:LINE:DEL?`

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:FREQuency <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:FREQuency ?

Description: The command sets the reference frequency for loss of the given device on the given port-pair of the LRL calibration on the given channel. The available port pair is 12 where it is fixed as Port 1 and Port 2. The command can be used by 2-Port and 4-Port VNA instruments. The query outputs the reference frequency for loss of the given device on the given port-pair of the LRL calibration on the given channel. The available port pair is 12.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 7E4 to 7E10

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT12:LINE:FREQ 1.0E9`

`:SENS1:CORR:COLL:LRL:DEV1:PORT12:LINE:FREQ?`

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LENGth <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LENGth?

Description: The command sets the effective (air-equivalent) line length of the given device on the given port-pair of the LRL calibration on the given channel. The available port pair is 12.

The query outputs the effective (air-equivalent) line length of the given device on the given port-pair of the LRL calibration on the given channel. The available port pair is 12.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Values: Dev X defaults of each band = 0

Dev Y defaults of each band:

Band 1: 5

Band 2: 4

Band 3: 3

Band 4: 2

Band 5: 1

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT12:LINE:LENG 2.0E-3`

`:SENS1:CORR:COLL:LRL:DEV1:PORT12:LINE:LENG?`

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LOSS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LOSS?

Description: Sets the line loss of the given device on the given port-pair of the LRL calibration on the given channel. The available port pair is 12. Outputs the line loss of the given device on the given port-pair of the LRL calibration on the given channel. The available port pair is 12.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: <NR3> The output parameter is in dB/mm.

Range: NA

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT12:LINE:LOSS 3.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT12:LINE:LOSS?`

:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:PLENgth <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:PLENgth?

Description: The command sets the physical line length of the given device of the LRL calibration on the given channel.

The query outputs the physical line length of the given device of the LRL calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: Default value is dependent on effective length default value.

Syntax Example: `:SENSe:CORR:COLL:LRL:DEV3:PORT12:LINE:PLEN 20E-3`

`:SENSe:CORR:COLL:LRL:DEV3:PORT12:LINE:PLEN?`

5-70 :SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs

The :SENSe{1-16}:CORRection:COLLect:LRL:PORT subsystem commands provide control of Line-Reflect-Line port assignments for FULL3 or FULL4 calibrations. The :LRL:PORT commands require a 4-Port VNA instrument.

LRL Calibration Subsystems

The LRL-related calibration commands are organized into five (5) subsystems in the following sections:

- ":SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem" on page 5-295
- ":SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem" on page 5-305
- ":SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs" on page 5-309
- ":SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs" on page 5-311
- ":SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem" on page 5-317

:SENSe{1-16}:CORRection:COLLect:LRL:PORT{13 | 14 | 23 | 24}:FULL3 <char>

Description: This command requires a 4-Port VNA instrument. Sets the calibration type to a Full 3-Port LRL for the indicated channel using either two 2-Port LRL Calibrations or one LRL 2-Port Calibration with One Singleton. No query.

Full 3-Port LRL Method #1 - Use Two 2-Port Calibrations

The first method requires two LRL FULL2 port calibrations where the calibrations share one common port. Port pairs 1-2 and 3-4 are not allowed. The two calibrations must share a common port or the command results in an error. Valid port pair combinations are:

- Port Pair 1-2 not allowed
- Port Pair 1-3 with Port Pair 1-4
- Port Pair 1-3 with Port Pair 2-3
- Port Pair 1-4 with Port Pair 1-3
- Port Pair 1-4 with Port Pair 2-4
- Port Pair 2-3 with Port Pair 2-4
- Port Pair 2-3 with Port Pair 1-3
- Port Pair 2-4 with Port Pair 1-4
- Port Pair 2-4 with Port Pair 2-3
- Port Pair 3-4 not allowed

The first calibration is referred to as :CALa; the second calibration referred to as :CALB.

The CALa port pair is assigned with the :PORT{13 | 14 | 23 | 24} keyword parameter.

The CALB port pair is assigned with the command <char> parameter with arguments of:

- PORT1 | PORT2 | PORT3 | PORT4 | PORT13 | PORT14 | PORT23 | PORT24

For example, the command for an LRL FULL3 calibration with the first LRL FULL2 (CALa) on port pair PORT13 and the second LRL FULL2 (CALB) on port pair PORT23 is shown below:

```
:SENS1:CORR:COLL:LRL:PORT13:FULL3 PORT23
```

Full 3-Port LRL Method #2 - Use One Two-Port Cal and One Singleton

The second way to perform an LRL FULL3 calibration is to include one LRL FULL2 calibration, a singleton reflection calibration on a different port. The method concludes with measuring a short or an open on the singleton port and one or two through lines on the port pair. Note that Port Pairs 1-2 and 3-4 are not allowed and the Port Pair assignments MUST NOT include the common singleton port. Valid port pairs, singleton, and through line combinations are:

- Port Pair 1-2 not allowed
- Port Pair 1-3 and Singleton 2 (and a through line cal between Ports 1-2 or 2-3)
- Port Pair 1-3 and Singleton 4 (and a through line cal between Ports 1-4 or 3-4)
- Port Pair 1-4 and Singleton 2 (and a through line cal between Ports 1-2 or 2-4)
- Port Pair 1-4 and Singleton 3 (and a through line cal between Ports 1-3 or 3-4)
- Port Pair 2-3 and Singleton 1 (and a through line cal between Ports 1-2 or 1-3)
- Port Pair 2-3 and Singleton 4 (and a through line cal between Ports 2-4 or 3-4)
- Port Pair 2-4 and Singleton 1 (and a through line cal between Ports 1-2 or 1-4)
- Port Pair 2-4 and Singleton 3 (and a through line cal between Ports 2-3 or 3-4)
- Port Pair 3-4 not allowed

An example of an LRL FULL2 calibration on port pair PORT23 and a singleton on PORT4 is shown below. The required through (thru) can be between port 2-4 or 3-4. The through line is thus configured between port 3 and 4:

```
:SENSe1:CORRection:COLLect:LRL:PORT23:FULL3 PORT4
:SENSe{1-16}:CORRections:COLLect:LRL:SINGleton:REFLection:TYPe
:SENSe(1-16):CORRection:COLLect:THRu:CLEar
:SENSe{1-16}:CORRection:COLLect:THRu:ADD PORT34
```

For additional information, also see the instrument menu for this method under:

- MAIN | Calibration | CALIBRATION | Calibrate | CALIBRATE | Manual Cal | MANUAL CAL | 3-Port Cal | THREE PORT CAL

From there, set the following:

- Cal Method = LRL/LRM
- Edit Cal Parameters | THREE PORT CAL SETUP | Cal Type = LRL/LRM+Singleton

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4 | PORT13 | PORT14 | PORT23 | PORT24

Range: NA

Default: PORT13

Syntax Example: `:SENS1:CORR:COLL:LRL:PORT23:FULL3 PORT13`

:SENSe{1-16}:CORRection:COLLect:LRL:PORT{13 | 14 | 23 | 24}:FULL4

Description: This command requires a 4-Port VNA instrument. Sets the calibration type to Full Four Port LRL for the indicated channel. In order to achieve a Full4 calibration using LRL techniques, it is necessary to perform two Full2 LRL calibrations on independent port pairs. The port number specified in this command is the port pair for the first LRL calibration. The port pair of the second LRL calibration is made up of the other ports. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:PORT13:FULL4`

5-71 :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs

The :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton subsystem commands provide configuration control of the Line-Reflect-Line calibration using singletons in different configurations. The commands provide control of singleton configuration parameters and output reporting. The use of the :LRL:SINGleton commands require a 4-Port VNA instrument.

LRL Calibration Subsystems

The LRL-related calibration commands are organized into five (5) subsystems in the following sections:

- Section 5-68 :SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem on page 5-295
- Section 5-69 :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem on page 5-305
- Section 5-70 :SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs on page 5-309
- Section 5-71 :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs on page 5-311 (this subsystem)
- Section 5-73 :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem on page 5-317

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:LOAD <string>

Description: This command loads an LRL cal kit file into the Singleton LRL calibration on the given channel.

Cmd Parameters: <string> Filename and path to the file in the form:

`x:\directory\filename.lcf`

where `x:\directory\filename.lcf` exist and `filename.lcf` is a valid LRL cal kit file.

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:SINGleton:CKIT:LOAD c:\filename.lcf`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:NAMe <char>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:NAMe?

Description: This command sets the name that will be stored in the Singleton LRL cal kit file.

The query outputs the name that was last set for the currently loaded file.

Cmd Parameters: <char> A name that will be associated with the current cal kit file.

Query Output: <char> The name that is currently associated with the current cal kit file.

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:SINGleton:CKIT:NAM mylrlcalkit`

`:SENS1:CORR:COLL:LRL:SINGleton:CKIT:NAM?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:SAVe <string>

Description: This command saves an LRL cal kit file into the Singleton calibration on the given channel.

Cmd Parameters: <string> Filename and path to the file in the form:

`x:\directory\filename.lcf`

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: `SENS1:CORR:COLL:LRL:SINGleton:CKIT:SAVe c:\filename.lcf`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C0?

Description: Sets the capacitance of the LRL singleton open on the given channel. The C0 (zero) coefficient is measured in Farads. Outputs the capacitance of the LRL singleton open on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: <NR3> The output parameter is in Farads.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:OPEN:C0 3.01E-12`

`:SENS1:CORR:COLL:LRL:SING:OPEN:C0?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C1?

Description: Sets the C1 (C one) coefficient of the LRL singleton open on the given channel. The C1 coefficient is measured in Farads/Hertz. Outputs the C1 coefficient of the LRL singleton open on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz.

Query Parameters: <NR3> The output parameter is in Farads/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:OPEN:C1 2.0E0`

`:SENS1:CORR:COLL:LRL:SING:OPEN:C1?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C2?

Description: Sets the C2 coefficient of the LRL singleton open on the given channel. The C2 coefficient is measured in Farads/Hertz^2. Outputs the C2 coefficient of the LRL singleton open on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^2 (Farads per Hertz squared).

Query Parameters: <NR3> The output parameter is in Farads/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:OPEN:C2 2.0E0`

`:SENS1:CORR:COLL:LRL:SING:OPEN:C2?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C3?

Description: Sets the C3 coefficient of the LRL singleton open on the given channel. The C3 coefficient is measured in Farads/Hertz^3. Outputs the C3 coefficient of the LRL singleton open on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^3 (Farads per Hertz cubed).

Query Parameters: <NR3> The output parameter is in Farads/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:OPEN:C3 2.0E0`

`:SENS1:CORR:COLL:LRL:SING:OPEN:C3?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:OFFSet <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:OFFSet?

Description: On 4-Port VNAs, sets the offset length of the LRL singleton open on the given channel for a three-port LRL calibration. A prior command defined the singleton as open or short. This command then defines open/short nature of that singleton. For port-pair and singleton LRL calibrations, the following combinations are allowed:

- Port Pair 1-2 may not be used.
- Port Pair 1-3 with singleton 2 or 4
- Port Pair 1-4 with singleton 2 or 3
- Port Pair 2-3 with singleton 1 or 2
- Port Pair 2-4 with singleton 1 or 3
- Port Pair 3-4 may not be used.

The query form outputs the offset length of the LRL singleton open on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:OPEN:OFFS 1.0E0`

`:SENS1:CORR:COLL:LRL:SING:OPEN:OFFS?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:REFLection:TYPe <char>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:REFLection:TYPe?

Description: Assigns the LRL Singleton reflection type on the given channel. Outputs the LRL Singleton reflection type on the given channel.

Cmd Parameters: <char> OPEN | SHORt

Query Parameters: <char> OPEN | SHOR

Range: NA

Default: OPEN

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:REFL:TYP SHOR`

`:SENS1:CORR:COLL:LRL:SING:REFL:TYP?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L0?

Description: Sets the inductance of the LRL singleton short on the given channel. The L0 (zero) coefficient is measured in Henrys. Outputs the inductance of the LRL singleton short on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:SHOR:L0 2.0E-6`

`:SENS1:CORR:COLL:LRL:SING:SHOR:L0?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L1?

Description: Sets the L1 (L one) coefficient of the LRL singleton short on the given channel. The L1 coefficient is measured in Henrys/Hertz. Outputs the L1 coefficient of the LRL singleton short on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SING:SHOR:L1 2.0E0`

`:SENS1:CORR:COLL:LRL:SING:SHOR:L1?`

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L2?

Description: Sets the L2 coefficient of the LRL singleton short on the given channel. The L2 coefficient is measured in Henrys/Hertz^2. Outputs the L2 coefficient of the LRL singleton short on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2 (Henrys per Hertz squared).

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example:
```
:SENS1:CORR:COLL:LRL:SING:SHOR:L2 2.0E0
:SENS1:CORR:COLL:LRL:SING:SHOR:L2?
```

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L3?

Description: The command sets the L3 coefficient of the LRL singleton short on the given channel. The L3 coefficient is measured in Henrys/Hertz^3. The query outputs the L3 coefficient of the LRL singleton short on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3 (Henrys per Hertz cubed).

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example:
```
:SENS1:CORR:COLL:LRL:SING:SHOR:L3 2.0E0
:SENS1:CORR:COLL:LRL:SING:SHOR:L3?
```

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:OFFSet <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:OFFSet?

Description: The command sets the offset length of the LRL singleton short on the given channel. The query outputs the offset length of the LRL singleton short on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example:
```
:SENS1:CORR:COLL:LRL:SING:SHOR:OFFS 1.0E0
:SENS1:CORR:COLL:LRL:SING:SHOR:OFFS?
```

5-72 :SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide Subsystem

The :SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide subsystem commands provide control of the waveguide Line-Reflect-Line parameters for dielectric and cutoff frequency values.

LRL Calibration Subsystems

The LRL-related calibration commands are organized into five (5) subsystems in the following sections:

- Section 5-68 :SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem on page 5-295
- Section 5-69 :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem on page 5-305
- Section 5-70 :SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs on page 5-309
- Section 5-71 :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs on page 5-311
- Section 5-73 :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem on page 5-317 (this subsystem)

:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:DIELectric <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:DIELectric?

Description: Sets the LRL waveguide calibration dielectric for the given channel. Outputs the LRL waveguide calibration dielectric for the given channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:WAV:DIEL <NRf>`

`:SENS1:CORR:COLL:LRL:WAV:DIEL?`

:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:FREQuency <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:FREQuency?

Description: The command sets the LRL waveguide calibration cutoff frequency for the given channel. Outputs the LRL waveguide calibration dielectric for the given channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Range: MPND

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:WAV:FREQ <NRf>`

`:SENS1:CORR:COLL:LRL:WAV:FREQ?`

5-73 :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem

The :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] subsystem commands provide control of Line-Reflect-Line first calibration (or CALa) configuration parameters, execution, and output reporting. The :CALa keyword is optional for 2-Port VNA instruments and is required for 4-Port VNA instruments.

LRL Calibration Subsystems

The LRL-related calibration commands are organized into five (5) subsystems in the following sections:

- Section 5-68 :SENSe{1-16}:CORRection:COLLect:LRL:CALB Subsystem on page 5-295
- Section 5-69 :SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10} Subsystem on page 5-305
- Section 5-70 :SENSe{1-16}:CORRection:COLLect:LRL:PORT Subsystem - 4-Port VNAs on page 5-309
- Section 5-71 :SENSe{1-16}:CORRection:COLLect:LRL:SINGleton Subsystem - 4-Port VNAs on page 5-311
- Section 5-73 :SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem on page 5-317 (this subsystem)

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND:COUNt <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND:COUNt?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the number of bands to use in the LRL CALa calibration on the given channel. Outputs the number of bands to use in the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is an integer.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 5

Default: 1

Syntax Example: `:SENS1:CORR:COLL:LRL:BAND:COUN 2`

`:SENS1:CORR:COLL:LRL:BAND:COUN?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND{1-5}:REFLection:TYPe <char>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND{1-5}:REFLection:TYPe?

Description: The use of the optional :CALa parameter requires a requires a 4-Port VNA instrument. Sets the Band reflection type of the LRL CALa calibration on the given channel. where:

- OPENlike = The reflection type is more like an open.
- SHORTlike = The reflection type is more like a short.
- BOTH = The reflection type is like both an open and a short.

Outputs the reflection type of the LRL CALa calibration on the given channel for a given band.

Cmd Parameters: <char> OPENlike | SHORTlike | BOTH

Query Output: <char> OPEN | SHORT | BOTH

Range: NA

Default: OPEN

Syntax Example: `:SENS1:CORR:COLL:LRL:BAND1:REFL:TYP OPEN`

`:SENS1:CORR:COLL:LRL:BAND1:REFL:TYP?`

:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:LOAD <string>

Description: This command loads an LRL cal kit file into CALa LRL calibration on the given channel.

Cmd Parameters: <string> Filename and path to the file in the form:

`x:\directory\filename.lcf`

where `x:\directory\filename.lcf` exists and filename`.lcf` is a valid LRL cal kit file.

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CKIT:LOAD 'c:\filename.lcf'`

:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:NAMe <char>
:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:NAMe?

Description: This command sets the name that will be stored in the LRL cal kit file.

The query outputs the name that was last set for the currently loaded file.

Cmd Parameters: <char> A name that will be associated with the current cal kit file.

Query Parameters: NA

Query Output: <char> The name that is currently associated with the current cal kit file.

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CKIT:NAM mylrlcalkit`

`:SENS1:CORR:COLL:LRL:CKIT:NAM?`

:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:SAVe <string>

Description: This command saves an LRL cal kit file into CALa LRL calibration on the given channel.

Cmd Parameters: <string> Filename and path to the file in the form:

x:\directory\filename.lcf

Query Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:LRL:CKIT:SAV c:\filename.lcf`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Initiates collection of the indicated Device Line Standard data for the CALa calibration on the indicated channel. No query.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:LINE`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:FREQuency <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:FREQuency?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the reference frequency for loss of the given device of the LRL CALa calibration on the given channel. Outputs the reference frequency for loss of the given device of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:LINE:FREQ 1.0E7`

`:SENS1:CORR:COLL:LRL:DEV1:LINE:FREQ?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LENGth <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LENGth?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the line length of the given device of the LRL CALa calibration on the given channel. Outputs the line length of the given device of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:LINE:LENG 1.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:LINE:LENG?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LOSS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LOSS?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the line loss of the given device of the LRL CALa calibration on the given channel. Outputs the line loss of the given device of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: <NR3> The output parameter is in dB/mm.

Range: MPND/1000 (MPND divided by 1000)

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:LINE:LOSS 3.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:LINE:LOSS?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C0?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets capacitance of the match device on the given port of the LRL CALa calibration on the given channel. The C0 (C zero) coefficient is in Farads. Outputs capacitance of the match device on the given port of the LRL CALa calibration on the given channel.

The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: <NR3> The output parameter is in Farads.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C0 3.01E-12`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C0?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C1?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the C1 (C one) coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The C1 coefficient is in Farads/Hertz.

Outputs the C1 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz.

Query Parameters: <NR3> The output parameter is in Farads/Hertz.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C1 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C1?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C2?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the C2 coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The C2 coefficient is in Farads/Hertz^2. Outputs the C2 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^2.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^2.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C2 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C2?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C3?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the C3 coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The C3 coefficient is in Farads/Hertz^3. Outputs the C3 coefficient of the match device on the given port of the LRL CALa calibration on the given channel

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^3.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^3.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C3 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:C3?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L0?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets inductance of the match device on the given port of the LRL CALa calibration on the given channel. The L0 (zero) coefficient is in Henrys. Outputs inductance of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L0 2.0E-6`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L0?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L1?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the L1 (L one) coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The L1 coefficient is in Henrys/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the L1 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L1`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L1?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L2?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the L2 coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The L2 coefficient is in Henrys/Hertz^2.

Outputs the L2 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L2 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L2?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L3?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the L3 coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The L3 coefficient is in Henrys/Hertz^3. Outputs the L3 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L3 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:L3?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF1 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF1?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the offset Length1 coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The OFF1 coefficient is measured in Meters/Hertz. Outputs the offset Length1 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz.

Query Parameters: <NR3> The output parameter is in Meters/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFF1 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFF1?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF2 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF2?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the offset Length2 coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The OFF2 coefficient is measured in Meters/Hertz^2. Outputs the offset Length2 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz^2.

Query Parameters: <NR3> The output parameter is in Meters/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFF2 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFF2?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF3 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF3?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets the offset Length3 coefficient of the match device on the given port of the LRL CALa calibration on the given channel. The OFF3 coefficient is measured in Meters/Hertz^3. Outputs the offset Length3 coefficient of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz^3.

Query Parameters: <NR3> The output parameter is in Meters/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFF3 2.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFF3?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFFS?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets offset length of the match device on the given port of the LRL CALa calibration on the given channel. Outputs offset length of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFFS 1.0E0`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:OFFS?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:R <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:R?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets resistance of the match device on the given port of the LRL CALa calibration on the given channel. Outputs resistance of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:R 7.5E1`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:R?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:Z0 <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:Z0?

Description: The use of the optional :CALa parameter, Port 3, and/or Port 4 require a 4-Port VNA instrument. Sets impedance of the match device on the given port of the LRL CALa calibration on the given channel. Outputs impedance of the match device on the given port of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:Z0 7.5E1`

`:SENS1:CORR:COLL:LRL:DEV1:PORT1:MATCH:Z0?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:TYPe <char>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:TYPe?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the type of the given device in the LRL CALa calibration on the given channel. Outputs the type of the given device in the LRL CALa calibration on the given channel.

Cmd Parameters: <char> LINE | MATCH | DEVICE1 | DEVICE2

Query Parameters: <char> LINE | MATCH | DEVICE1 | DEVICE2

Range: NA

Default: LINE

Syntax Example: `:SENS1:CORR:COLL:LRL:DEV1:TYP LINE`

`:SENS1:CORR:COLL:LRL:DEV1:TYP?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the breakpoint frequency of the LRL CALa calibration on the given channel. Outputs the breakpoint frequency of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default: 3.00000000000E+009

Syntax Example: `:SENS1:CORR:COLL:LRL:FREQ:BRE 1.0E7`

`:SENS1:CORR:COLL:LRL:FREQ:BRE?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint{1-4} <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint{1-4}?

Description: The command sets the breakpoint frequency between bands of LRL calibration on a given channel.

The query outputs the breakpoint frequency between bands of LRL calibration on a given channel. For a 4-port system, the command is meant for CAL A of LRL.

In the command, the number next to BREakpoint has the following correspondence:

- BREakpoint1: Breakpoint frequency between bands 2 and 1
- BREakpoint2: Breakpoint frequency between bands 3 and 2
- BREakpoint3: Breakpoint frequency between bands 4 and 3
- BREakpoint1: Breakpoint frequency between bands 5 and 4

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default: 3E09

Syntax Example: `:SENS:CORR:COLL:LRL:FREQ:BRE1 4E09`

`:SENS:CORR:COLL:LRL:FREQ:BRE1?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:OPEN:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:OPEN:OFFS?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the offset length of the open like reflection of the LRL CALa calibration on the given channel. Outputs the offset length of the open like reflection of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:OPEN:OFFS 1.0E0`

`:SENS1:CORR:COLL:LRL:OPEN:OFFS?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:REFPlane <char>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:REFPlane?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the location of the reference plane in the LRL CALa calibration on the given channel where:

- MIDdle = the middle of the transmission line
- END = the end of the transmission line

Outputs the location of the reference plane in the LRL CALa calibration on the given channel.

Cmd Parameters: <char> MIDdle | END

Query Parameters: <char> MID | END

Range: NA

Default: END

Syntax Example: `:SENS1:CORR:COLL:LRL:REFP MID`

`:SENS1:CORR:COLL:LRL:REFP?`

:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:SHORT:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:SHORT:OFFS?

Description: The use of the optional :CALa parameter requires a 4-Port VNA instrument. Sets the offset length of the short like reflection of the LRL CALa calibration on the given channel. Outputs the offset length of the short like reflection of the LRL CALa calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:LRL:SHORT:OFFS 1.0E0`

`:SENS1:CORR:COLL:LRL:SHORT:OFFS?`

5-74 :SENSe{1-16}:CORRection:COLLect:METHod Subsystem

The :SENSe{1-16}:CORRection:COLLect:METHod subsystem command sets the calibration method for the indicated channel. When configuring the instrument calibration this option must be set first, generally followed by line type, and finally calibration type. Optionally, these commands are followed by coefficient and characterization commands for user-defined calibration devices and kits.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:COLLect:METHod <char>
:SENSe{1-16}:CORRection:COLLect:METHod?

Description: The command sets the calibration method for the indicated channel. The following calibration methods are available:

- AUTOcal = Automatic Calibrator (AutoCal) Module calibration method using Anritsu 36585-Series Precision AutoCal Modules
- ACLIght = Automatic Calibrator (AutoCal) Module calibration method using Anritsu 36581-Series Standard AutoCal Modules.
- LRL = Line-Reflect-Line calibration method
- LRM = Line-Reflect-Match calibration method
- SOLR = Short-Open-Load-Reciprocal calibration method
- SOLT = Short-Open-Load-Through calibration method
- SSLT = Short-Short-Load-Through calibration method
- SSST = Short-Short-Short-Through calibration method

The query outputs the calibration method for the indicated channel.

Cmd Parameters: <char> AUTOcal | ACLIght | LRL | LRM | SOLR | SOLT | SSLT | SSST

Query Parameters: <char> AUTO | ACLI | LRL | LRM | SOLR | SOLT | SSLT | SSST

Range: NA

Default Value: SOLT

Syntax Example: `:SENS1:CORR:COLL:METH AUTO`

`:SENS1:CORR:COLL:METH?`

5-75 :SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem

The :SENSe{1-16}:CORRection:COLLect:MICrostrip subsystem commands set the parameter values for dielectric, kit type, and port assigned for microstrip substrate values.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:COLLect:MICrostrip:DIELectric <NRf>
:SENSe{1-16}:CORRection:COLLect:MICrostrip:DIELectric?

Description: Sets the microstrip substrate dielectric value for calibration on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their command parameters. Outputs the microstrip substrate dielectric value for calibration on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Query Parameters.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example:
```
:SENS1:CORR:COLL:MIC:DIEL 1.2E0
:SENS1:CORR:COLL:MIC:DIEL?
```

:SENSe{1-16}:CORRection:COLLect:MICrostrip:EFFective <NRf>
:SENSe{1-16}:CORRection:COLLect:MICrostrip:EFFective?

Description: Sets the microstrip effective dielectric value for calibration on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters. Outputs the microstrip effective dielectric value for calibration on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:MIC:EFF 1.2E0`

`:SENS1:CORR:COLL:MIC:EFF?`

:SENSe{1-16}:CORRection:COLLect:MICrostrip:KIT <char>
:SENSe{1-16}:CORRection:COLLect:MICrostrip:KIT?

Description: Selects the microstrip kit to use for calibration on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Available Microstrip Kits

The available microstrip kit parameters are:

- MIL10 = Standard 10-mil (0.010” or 0.25400 mm thick) microstrip
- MIL15 = Standard 15-mil (0.015” or 0.38100 mm thick) microstrip
- MIL25 = Standard 10-mil (0.025” or 0.63500 mm thick) microstrip
- USERx = User-defined microstrip 1 through 32 (e.g., USER5, USER23)

Using User-Defined Microstrip Kits

User-defined microstrips (USER1 through USER8 above) can be defined through the menu-driven user interface by entering six values:

- Microstrip Kit Label = Defaults as “User-DefinedN” (where N = 1 to 8) and can be changed as required. Programmatically, the each user-defined microstrip kit name must be still referred to as the appropriate “USERn” parameter.
- Strip Width (mm). Programmatically, the width is entered in Meters.
- Impedance (Ohms)
- Substrate Thickness (mm). Programmatically, the thickness is entered in Meters.
- Substrate Dielectric Value
- Effective Dielectric Value

Query Output

Outputs the microstrip kit selected for calibration on the indicated channel.

See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <char> MIL10 | MIL15 | MIL25 | USER1 | USER2 | USER3 | USER4 | USER5 | USER6 | USER7 | USER8 | USER9 | USER10 | USER11 | USER12 | USER13 | USER14 | USER15 | USER16 | USER17 | USER18 | USER19 | USER20 | USER21 | USER22 | USER23 | USER24 | USER25 | USER26 | USER27 | USER28 | USER29 | USER30 | USER31 | USER32

Query Parameters: <char> MIL10 | MIL15 | MIL25 | USER1 | USER2 | USER3 | USER4 | USER5 | USER6 | USER7 | USER8 | USER9 | USER10 | USER11 | USER12 | USER13 | USER14 | USER15 | USER16 | USER17 | USER18 | USER19 | USER20 | USER21 | USER22 | USER23 | USER24 | USER25 | USER26 | USER27 | USER28 | USER29 | USER30 | USER31 | USER32

Range: NA

Default Value: MIL10

Syntax Example:
```
:SENS1:CORR:COLL:MIC:KIT MIL15
:SENS1:CORR:COLL:MIC:KIT?
```

:SENSe{1-16}:CORRection:COLLect:MICrostrip:PORT{1-4}:CONNector <char>
:SENSe{1-16}:CORRection:COLLect:MICrostrip:PORT{1-4}:CONNector?

Description: Sets the microstrip kit connector type for the indicated port on the indicated channel to where only user-defined microstrips can be used in the <char> parameter as:

- USERx = User-defined microstrip 1 through 32 (e.g., USER5, USER23)

See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their command parameters.

User-Defined Microstrips

In the menu-driven user interface, user-defined microstrips are set with six values:

- Microstrip Kit Label = Defaults as “User-DefinedN” (where N = 1 to 8) and can be changed as required.
- Strip Width (mm). Programmatically, the width is entered in Meters.
- Impedance (Ohms)
- Substrate Thickness (mm). Programmatically, the thickness is entered in Meters.
- Substrate Dielectric Value
- Effective Dielectric Value

The query outputs the microstrip kit connector type for the indicated port on the indicated channel. See Table 2-22, “Connector Type Abbreviations and Descriptions” on page 2-48 for a complete listing of calibration components, connectors, and their parameters.

Cmd Parameters: <char> MIL10 | MIL15 | MIL25 | USER1 | USER2 | USER3 | USER4 | USER5 | USER6 | USER7 | USER8 | USER9 | USER10 | USER11 | USER12 | USER13 | USER14 | USER15 | USER16 | USER17 | USER18 | USER19 | USER20 | USER21 | USER22 | USER23 | USER24 | USER25 | USER26 | USER27 | USER28 | USER29 | USER30 | USER31 | USER32

Query Parameters: <char> MIL10 | MIL15 | MIL25 | USER1 | USER2 | USER3 | USER4 | USER5 | USER6 | USER7 | USER8 | USER9 | USER10 | USER11 | USER12 | USER13 | USER14 | USER15 | USER16 | USER17 | USER18 | USER19 | USER20 | USER21 | USER22 | USER23 | USER24 | USER25 | USER26 | USER27 | USER28 | USER29 | USER30 | USER31 | USER32

Range: NA

Default Value: CMV

Syntax Example: `:SENS1:CORR:COLL:MIC:PORT1:CONN USER2`

`:SENS1:CORR:COLL:MIC:PORT1:CONN?`

:SENSe{1-16}:CORRection:COLLect:MICrostrip:THICKness <NRf>
:SENSe{1-16}:CORRection:COLLect:MICrostrip:THICKness?

Description: Sets the microstrip substrate thickness for calibration on the indicated channel.

Outputs the microstrip substrate thickness for calibration on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:MIC:THICK 1.2E-5`

`:SENS1:CORR:COLL:MIC:THICK?`

:SENSe{1-16}:CORRection:COLLect:MICrostrip:WIDth <NRf>
:SENSe{1-16}:CORRection:COLLect:MICrostrip:WIDth?

Description: Sets the microstrip width for calibration on the indicated channel.

Outputs the microstrip width for calibration on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:MIC:WID 1.2E-4`

`:SENS1:CORR:COLL:MIC:WID?`

:SENSe{1-16}:CORRection:COLLect:MICrostrip:Z0 <NRf>
:SENSe{1-16}:CORRection:COLLect:MICrostrip:Z0?

Description: Sets the microstrip impedance (Z zero) for calibration on the indicated channel. Outputs the microstrip impedance for calibration on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:MIC:Z0 7.5E1`

`:SENS1:CORR:COLL:MIC:Z0?`

5-76 :SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem

The:SENSe{1-16}:CORRection:COLLect:MULTIple subsystem command sets transmission through (thru) lines between one or more port pairs.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:COLLect:MULTiple:THRu <char> {<char>,<char>,<char>,<char>,<char>}
:SENSe{1-16}:CORRection:COLLect:MULTiple:THRu?

Description: The command adds one or more Transmission Throughs (Thrus) to the calibration process on the indicated channel. The use of Port 3 and/or Port 4 on a through requires a 4-Port VNA instrument. The available throughs are:

- THRu12 = Through line between Port 1 and Port 2. The short-form is THR12.
- THRu13 = Through line between Port 1 and Port 3. The short-form is THR13.
- THRu14 = Through line between Port 1 and Port 4. The short-form is THR14.
- THRu23 = Through line between Port 2 and Port 3. The short-form is THR23.
- THRu24 = Through line between Port 2 and Port 4. The short-form is THR24.
- THRu34 = Through line between Port 3 and Port 4. The short-form is THR34.

Outputs the calibration process list of Transmission Throughs on the indicated channel

Cmd Parameters: <char> THRu12 | THRu13 | THRu14 | THRu23 | THRu24 | THRu34

Query Parameters: <char> THR12 | THR13 | THR14 | THR23 | THR24 | THR34

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:MULT:THR THR12, THR23, THR24`

`:SENS1:CORR:COLL:MULT:THR?`

5-77 :SENSe{1-16}:CORRection:COLLect:PORT Subsystem

The :SENSe{1-16}:CORRection:COLLect:PORT subsystem commands start an actual instrument calibration.

Calibration Subsystems with Actual Calibrations

Related calibration subsystems that perform actual calibrations are:

- ":SENSe{1-16}:ABORtcal Subsystem" on page 5-264
- ":SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs" on page 5-279
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- ":SENSe{1-16}:CORRection:COLLect:FULL4 Subsystem - 4-Port VNAs" on page 5-289
- ":SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem" on page 5-317
- ":SENSe{1-16}:CORRection:COLLect:PORT Subsystem" on page 5-336
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380

Calibration Type Abbreviations

The commands in this subsystem use the following abbreviations for different calibration methods:

- 1P2PF = One-path two-port calibration, forward direction
- 1P2PR = One path two port calibration, reverse direction
- FULL1 = Full one port calibration
- FULL2 = Full two port calibration
- FULLB = Full one port reflection calibration, both ports
- RESP1 = One port response calibration
- RESPB = One port response calibration, both ports
- TFRB = Transmission frequency response calibration, both directions
- TFRF = Transmission frequency response calibration, forward direction
- TFRR = Transmission frequency response calibration, reverse direction

:SENSe{1-16}:CORRection:COLLect:PORT{1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34 | 123 | 124 | 134 | 234 | 1234}:FULL1

Description: Selects Full One-Port Calibration as the calibration type on the indicated channel and ports selected. The Full 1 port calibration can be performed on any combination of ports by listing them using as PORTw | wx | wxy | wxyz where:

- w = the first port
- x = the second port
- y = the third port
- z = the fourth port

The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

For example, to perform a FULL1 calibration on Channel 1 and Port 1, 2, and 4, the command is:

```
SENS1:CORR:COLL:PORT124:FULL1
```

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT23:FULL1`

:SENSe{1-16}:CORRection:COLLect:PORT{1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34 | 123 | 124 | 134 | 234 | 1234}:RESP1

Description: Selects One-Port Response Calibration as the calibration type on the indicated channel and ports selected. The response calibration can be performed on one or more ports by listing them using as PORTw | wx | wxy | wxyz where:

- w = the first port
- x = the second port
- y = the third port
- z = the fourth port

For example, to perform a response calibration on Channel 1 and Ports 1, 2, and 3, the command is:

```
SENS1:CORR:COLL:PORT123:RESP1
```

The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT24:RESP1`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:ISOL

Description: Initiates collection of the isolation standard data for the calibration on the given channel. For a 2-Port VNA, the available port pair is 12. Port pairs using Port 3 and/or Port 4 require a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT12:ISOL`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRB

Description: Adds Transmission Frequency Response Calibration Both directions on the indicated Channel and port pair. The Transmission Frequency Response calibration can be any combination of port pairs, and any combination of Forward only (TFRF), Reverse only (TFRR) or both (TFRB). This command adds a Transmission Frequency Response Calibration for Both directions. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT13:TFRB`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRF

Description: Adds Transmission Frequency Response Calibration Forward direction on the indicated Channel and port pair. The Transmission Frequency Response calibration can be any combination of port pairs, and any combination of Forward only (TFRF), Reverse only (TFRR) or both (TFRB). This command adds a Transmission Frequency Response Calibration Forward direction. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT24:TFRF`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRR

Description: Adds Transmission Frequency Response Calibration Reverse direction on the indicated Channel and port pair. The Transmission Frequency Response calibration can be any combination of port pairs, and any combination of Forward only (TFRF), Reverse only (TFRR) or both (TFRB). This command adds a Transmission Frequency Response Calibration Reverse direction. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT12:TFRR`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu

Description: Initiates collection of the through (thru) standard data for the calibration on the given channel. For 2-Port VNAs, the available port pair is 12. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:FREQuency <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:FREQuency?

Description: Sets the through-line (thru-line) reference frequency of the loss on the selected port-pair of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the thru-line reference frequency of the loss on the selected port-pair of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR:FREQ 1.0E6`

`:SENS1:CORR:COLL:PORT12:THR:FREQ?`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LABEL?

Description: Sets the through-line (thru-line) label on the selected port-pair of the indicated channel. On 3-Port VNAs, the available port pair is 12. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the thru-line label on the selected port-pair of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

See definition of “<char>” on page 2-14.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT12:THR:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LENGth <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LENGth?

Description: Sets the through-line (thru-line) length on the selected port-pair of the indicated channel. The port-pair for 2-Port VNAs is Port 1-2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the thru-line length on the selected port-pair of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR:LENG 1.0E0`

`:SENS1:CORR:COLL:PORT12:THR:LENG?`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LOSS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LOSS?

Description: Sets the through-line (thru-line) loss on the selected port-pair of the indicated channel. The port-pair for 2-Port VNAs is Port 1-2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the thru-line loss on the selected port-pair of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: <NR3> The output parameter is in dB/mm.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR:LOSS 3.0E0`

`:SENS1:CORR:COLL:PORT12:THR:LOSS?`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal <char>
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal?

Description: Sets the through-line (thru-line) use reciprocal flag on the selected port-pair of the indicated channel. The port-pair for 2-Port VNAs is Port 1-2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the thru-line use reciprocal flag on the selected port-pair of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA>

Default Value: 0

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR:RECIP ON`

`:SENS1:CORR:COLL:PORT12:THR:RECIP?`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:SERial?

Description: Sets the through-line (thru-line) serial number on the selected port-pair of the indicated channel. The port-pair for 2-Port VNAs is Port 1-2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the thru-line serial number on the selected port-pair of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters. See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR:SER '123456'`

`:SENS1:CORR:COLL:PORT12:THR:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:Z0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:Z0?

Description: Sets the through-line (thru-line) impedance on the selected port-pair of the indicated channel. The port-pair for 2-Port VNAs is Port 1-2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the thru-line impedance on the selected port-pair of the indicated channel. The port-pair for 2-Port VNAs is Port 1-2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: 1E-4 to 1E10

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT12:THR:Z0 7.5E1`

`:SENS1:CORR:COLL:PORT12:THR:Z0?`

:SENSe{1-16}:CORRection:COLLect:PORT{123 | 124 | 134 | 234}:FULL3

Description: Sets the calibration type to Full Three Port for the indicated channel and port triplet where:

- 123 = Port Triplet of Port 1, Port 2, and Port 3
- 124 = Port Triplet of Port 1, Port 2, and Port 4
- 134 = Port Triplet of Port 1, Port 3, and Port 4
- 234 = Port Triplet of Port 2, Port 3, and Port 4

This command requires a 4-Port VNA instrument.

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT234:FULL3`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:CONNector <char>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:CONNector?

Description: Sets the connector type for the indicated port on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. The connector types are:

- CM2-V = Anritsu 2.4 mm (male)
- CM2 = 2.4 mm (male)
- CM3 = GPC 3.5 (male)
- CMK = K (male)
- CMN = K (male)
- CMS = SMA (male)
- CMC = TNC (male)
- CMV = V (male)
- CM1 = 1 mm (male)
- CM0.8 = 0.8mm (male)
- CM716 = 7/16 (male)
- CNG = GPC7 (none)
- CMN75 = N 75 Ohm (male)
- CMUn = User-defined male connector from 1 to 32, where: CMU1 = User-defined male 1, CMU2 = User-defined male 2, CMU3 = User-defined male 3, etc.
- CF2-V = Anritsu 2.4 mm (female)
- CF2 = 2.4 mm (female)
- CF3 = GPC 3.5 (female)
- CFK = K (female)
- CFN = N (female)
- CFS = SMA (female)
- CFC = TNC (female)
- CFV = V (female)
- CF1 = 1 mm (female)
- CF0.8 = 0.8mm (female)
- CF716 = 7/16 (female)
- CFN75 = N 75 Ohm (female)
- CFUn = User-defined female connector from 1 to 32, where: CFU1 = User-defined female 1, CFU2 = User-defined female 2, CFU3 = User-defined female 3, etc.

Outputs the connector type for the indicated port on the indicated channel.

See Table 2-22, “Connector Type Abbreviations and Descriptions” on page 2-48 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <char> CM1 | CMV | CMK | CM2 | CM2-V | CM3 | CMS | CMN | CMN75 | CNG | CM716 | CMC | CM0.8 | CMU1 | CMU2 | CMU3 | CMU4 | CMU5 | CMU6 | CMU7 | CMU8 | CMU9 | CMU10 | CMU11 | CMU12 | CMU13 | CMU14 | CMU15 | CMU16 | CMU17 | CMU18 | CMU19 | CMU20 | CMU21 | CMU22 | CMU23 | CMU24 | CMU25 | CMU26 | CMU27 | CMU28 | CMU29 | CMU30 | CMU31 | CMU32 | CF1 | CFV | CFK | CF2 | CF2-V | CF3 | CFS | CFN | CFN75 | CF716 | CFC | CF0.8 | CFU1 | CFU2 | CFU3 | CFU4 | CFU5 | CFU6 | CFU7 | CFU8 | CFU9 | CFU10 | CFU11 | CFU12 | CFU13 | CFU14 | CFU15 | CFU16 | CFU17 | CFU18 | CFU19 | CFU20 | CFU21 | CFU22 | CFU23 | CFU24 | CFU25 | CFU26 | CFU27 | CFU28 | CFU29 | CFU30 | CFU31 | CFU32

Query Parameters: <char> CM1 | CMV | CMK | CM2 | CM2-V | CM3 | CMS | CMN | CMN75 | CNG | CM716 | CMC | CM0.8 | CMU1 | CMU2 | CMU3 | CMU4 | CMU5 | CMU6 | CMU7 | CMU8 | CMU9 | CMU10 | CMU11 | CMU12 | CMU13 | CMU14 | CMU15 | CMU16 | CMU17 | CMU18 | CMU19 | CMU20 | CMU21 | CMU22 | CMU23 | CMU24 | CMU25 | CMU26 | CMU27 | CMU28 | CMU29 | CMU30 | CMU31 | CMU32 | CF1 | CFV | CFK | CF2 | CF2-V | CF3 | CFS | CFN | CFN75 | CF716 | CFC | CF0.8 | CFU1 | CFU2 | CFU3 | CFU4 | CFU5 | CFU6 | CFU7 | CFU8 | CFU9 | CFU10 | CFU11 | CFU12 | CFU13 | CFU14 | CFU15 | CFU16 | CFU17 | CFU18 | CFU19 | CFU20 | CFU21 | CFU22 | CFU23 | CFU24 | CFU25 | CFU26 | CFU27 | CFU28 | CFU29 | CFU30 | CFU31 | CFU32

Range: NA

Default Value: CMV

Syntax Example: `:SENS1:CORR:COLL:PORT1:CONN CM2`

`:SENS1:CORR:COLL:PORT1:CONN?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD

Description: Initiates collection of the Load Standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:SELect <char>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:SELect?

Description: Selects the load standard of LOAD1 or LOAD2 to be used for calibration on the indicated channel and port. Outputs the selected load standard of LOAD1 or LOAD2 to be used for calibration on the indicated channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

Cmd Parameters: <char> LOAD1 | LOAD2

Query Parameters: <char> LOAD1 | LOAD2

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD:SEL LOAD2`

`:SENS1:CORR:COLL:PORT1:LOAD:SEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:TYPe <char>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:TYPe?

Description: Selects the Broadband Load Type as Fixed or Sliding for calibration on the indicated channel and port. Outputs the Broadband Load Type selection on the indicated channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

Cmd Parameters: <char> FIXed | SLIDing

Query Parameters: <char> FIX | SLID

Range: NA

Default: FIX

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD:TYP FIX`

`:SENS1:CORR:COLL:PORT1:LOAD:TYP?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C0?

Description: Sets the Load1 standard C0 (C zero) capacitance on the selected port of the indicated channel. The C0 coefficient is measured in Farads. Outputs the Load1 standard C0 capacitance on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Farads.

Cmd Parameters: <NR3> The output parameter is in Farads.

Range: 0 to 1E12

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:C0 3.0E-11`

`:SENS1:CORR:COLL:PORT1:LOAD1:C0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C1?

Description: Sets the Load1 Standard C1 (C one) Coefficient on the selected port of the indicated channel. The C1 coefficient is measured in Farads/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 Standard C1 Coefficient on the Selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz.

Query Parameters: <NR3> The output parameter is in Farads/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:C1 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:C1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C2?

Description: Sets the Load1 Standard C2 Coefficient on the selected port of the indicated channel. The C2 coefficient is measured in Farads/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 Standard C2 Coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^2 (Hertz E2).

Query Parameters: <NR3> The output parameter is in Farads/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:C2 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:C2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C3?

Description: Sets the Load1 Standard C3 Coefficient on the selected port of the indicated channel. The C3 coefficient is measured in Farads/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 Standard C3 Coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^3 (Hertz E3).

Query Parameters: <NR3> The output parameter is in Farads/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:C3 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:C3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L0?

Description: Sets the Load1 standard L0 (L zero) inductance on the selected port of the indicated channel. The L0 coefficient is measured in Henrys. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard L0 inductance on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:L0 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:L0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L1?

Description: Sets the Load1 standard L1 (L one) inductance coefficient on the selected port of the indicated channel. The L1 coefficient is measured in Henrys/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard L1 inductance coefficient on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:L1 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:L1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L2?

Description: Sets the Load1 standard L2 inductance coefficient on the selected port of the indicated channel. The L2 coefficient is measured in Henrys’/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard L2 inductance coefficient on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:L2 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:L2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L3?

Description: Sets the Load1 standard L3 inductance coefficient on the selected port of the indicated channel. The L3 coefficient is measured in Henrys/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard L3 inductance coefficient on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:L3 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:L3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:LABEL?

Description: Sets the Load1 standard label on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard label on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT1:LOAD1:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF1?

Description: Sets the Load1 Standard Offset length1 coefficient on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 Standard Offset length1 coefficient on the Selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz.

Query Parameters: <NR3> The output parameter is in Meters/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:OFF1 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:OFF1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF2?

Description: Sets the Load1 Standard Offset length2 coefficient on the Selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 Standard Offset length2 coefficient on the Selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz^2.

Query Parameters: <NR3> The output parameter is in Meters/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:OFF2 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:OFF2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF3?

Description: Sets the Load1 Standard Offset length3 coefficient on the Selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 Standard Offset length3 coefficient on the Selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz^3.

Query Parameters: <NR3> The output parameter is in Meters/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:OFF3 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD1:OFF3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFFS?

Description: Sets the Load1 (Load one) standard offset on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard offset on the selected port of the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:OFFS 2.5E1`

`:SENS1:CORR:COLL:PORT1:LOAD1:OFFS?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:R <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:R?

Description: Sets the Load1 (Load one) standard resistance on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard resistance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: 0 to 1E10

Default Value: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:R 7.5E1`

`:SENS1:CORR:COLL:PORT1:LOAD1:R?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:SERial?

Description: Sets the Load1 (Load one) standard serial number on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard serial number on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:SER '123456'`

`:SENS1:CORR:COLL:PORT1:LOAD1:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:Z0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:Z0?

Description: Sets the Load1 (Load one) standard Impedance on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load1 standard Impedance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: 1E-4 to 1E10

Default Value: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD1:Z0 7.5E1`

`:SENS1:CORR:COLL:PORT1:LOAD1:Z0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C0?

Description: Sets the Load2 standard C0 (C zero) capacitance on the selected port of the indicated channel. The C0 coefficient is measured in Farads. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard C0 capacitance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: <NR3> The output parameter is in Farads.

Range: 0 to 1E12

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:C0 2.0E-11`

`:SENS1:CORR:COLL:PORT1:LOAD2:C0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C1?

Description: Sets the Load2 Standard C1 (C one) Coefficient on the Selected port of the indicated channel. The C1 coefficient is measured in Farads/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 Standard C1 Coefficient on the Selected port of the indicated channel

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz.

Query Parameters: <NR3> The output parameter is in Farads/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:C1 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:C1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C2?

Description: Sets the Load2 Standard C2 Coefficient on the selected port of the indicated channel. The C2 coefficient is measured in Farads/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 Standard C2 Coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^2.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:C2 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:C2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C3?

Description: Sets the Load2 Standard C3 Coefficient on the selected port of the indicated channel. The C3 coefficient is measured in Farads/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 Standard C3 Coefficient on the Selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^3.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:C3 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:C3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L0?

Description: Sets the Load2 standard L0 (L zero) inductance on the selected port of the indicated channel. The L0 coefficient is measured in Henrys. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard L0 inductance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:L0 2.0E-6`

`:SENS1:CORR:COLL:PORT1:LOAD2:L0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L1?

Description: Sets the Load2 standard L1 (L one) inductance coefficient on the selected port of the indicated channel. The L1 coefficient is measured in Henrys/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard L1 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:L1 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:L1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L2?

Description: Sets the Load2 standard L2 inductance coefficient on the selected port of the indicated channel. The L2 coefficient is measured in Henrys/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard L2 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:L2 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:L2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L3?

Description: Sets the Load2 standard L3 inductance coefficient on the selected port of the indicated channel. The L3 coefficient is measured in Henrys/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard L3 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:L3 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:L3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:LABEL?

Description: Sets the Load2 standard label on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard label on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT1:LOAD2:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF1?

Description: Sets the Load2 Standard Offset length1 (length one) coefficient on the selected port of the indicated channel. The OFF1 coefficient is measured in Meters/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 Standard Offset length1 coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz.

Query Parameters: <NR3> The output parameter is in Meters/Hertz.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:OFF1 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:OFF1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF2?

Description: Sets the Load2 Standard Offset length2 coefficient on the selected port of the indicated channel. The OFF2 coefficient is measured in Meters/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 Standard Offset length2 coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz^2.

Query Parameters: <NR3> The output parameter is in Meters/Hertz^2.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:OFF2 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:OFF2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF3?

Description: Sets the Load2 Standard Offset length3 coefficient on the selected port of the indicated channel. The OFF3 coefficient is measured in Meters/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 Standard Offset length3 coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters/Hertz^3.

Query Parameters: <NR3> The output parameter is in Meters/Hertz^3.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:OFF3 2.0E0`

`:SENS1:CORR:COLL:PORT1:LOAD2:OFF3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFFS?

Description: Sets the Load2 standard offset on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard offset on the selected port of the indicated channel.

Query Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:OFFS 2.5E1`

`:SENS1:CORR:COLL:PORT1:LOAD2:OFFS?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:R <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:R?

Description: Sets the Load2 standard resistance on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard resistance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: 0 to 1E10

Default Value: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:R 7.5E1`

`:SENS1:CORR:COLL:PORT1:LOAD2:R?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:SERial?

Description: Sets the Load2 standard serial number on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard serial number on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:SER '123456'`

`:SENS1:CORR:COLL:PORT1:LOAD2:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:Z0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:Z0?

Description: Sets the Load2 standard Impedance on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Load2 standard Impedance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: 1E-4 to 1E10

Default Value: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:PORT1:LOAD2:Z0 7.5E1`

`:SENS1:CORR:COLL:PORT1:LOAD2:Z0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN

Description: Initiates collection of the open standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C0?

Description: Sets the open standard C0 (C zero) capacitance on the selected port of the indicated channel. The C0 coefficient is measured in Farads. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the open standard C0 capacitance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: <NR3> The output parameter is in Farads.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN:C0 2.0E-11`

`:SENS1:CORR:COLL:PORT1:OPEN:C0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C1?

Description: Sets the open standard C1 (C one) capacitance coefficient on the selected port of the indicated channel. The C1 coefficient is measured in Farads/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the open standard C1 capacitance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz.

Query Parameters: <NR3> The output parameter is in Farads/Hertz.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN:C1 2.0E0`

`:SENS1:CORR:COLL:PORT1:OPEN:C1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C2?

Description: Sets the open standard C2 capacitance coefficient on the selected port of the indicated channel. The C2 coefficient is measured in Farads/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the open standard C2 capacitance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^2.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^2.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN:C2 2.0E0`

`:SENS1:CORR:COLL:PORT1:OPEN:C2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C3?

Description: Sets the open standard C3 capacitance coefficient on the selected port of the indicated channel. The C3 coefficient is measured in Farads/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the open standard C3 capacitance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^3.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^3.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN:C3 2.0E0`

`:SENS1:CORR:COLL:PORT1:OPEN:C3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:LABEL?

Description: Sets the open standard label on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the open standard label on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT1:OPEN:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:OFFS?

Description: Sets the open standard offset on the selected port of the indicated channel. The OFFS parameter is measured in Meters. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the open standard offset on the selected port of the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN:OFFS 2.5E1`

`:SENS1:CORR:COLL:PORT1:OPEN:OFFS?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:SERial?

Description: Sets the open standard serial number on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the open standard serial number on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters. See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:OPEN:SER '123456'`

`:SENS1:CORR:COLL:PORT1:OPEN:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:REFLection:COMPonent <char>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:REFLection:COMPonent?

Description: Selects the reflection standard to use for the reflection calibration on the indicated channel and port where the reflection standard types can be:

- OPEN = Open
- SHORt = Short
- OFSH1 = Offset Short 1
- OFSH2 = Offset Short 2
- OFSH3 = Offset Short 3
- NONE = Only returned by the query if not reflection component has been defined.

The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the reflect standard to use for the reflection calibration on the indicated channel and port.

Cmd Parameters: <char> OPEN | SHORt | OFSH1 | OFSH2 | OFSH3

Query Parameters: <char> OPEN | SHOR | OFSH1 | OFSH2 | OFSH3 | NONE

Range: NA

Default: OPEN

Syntax Example: `:SENS1:CORR:COLL:PORT1:REFL:COMP OPEN`

`:SENS1:CORR:COLL:PORT1:REFL:COMP?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT

Description: Initiates collection of the short standard data for the calibration on the given channel and port. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their command parameters. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L0?

Description: Sets the short standard L0 (L zero) inductance on the selected port of the indicated channel. The L0 coefficient is measured in Henrys. Outputs the short standard L0 inductance on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT:L0 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT:L0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L1?

Description: Sets the short standard L1 (L one) inductance coefficient on the selected port of the indicated channel. The L1 coefficient is measured in Henrys/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the short standard L1 inductance coefficient on the selected port of the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their command parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT:L1 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT:L1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L2?

Description: Sets the short standard L2 inductance coefficient on the selected port of the indicated channel. The L2 coefficient is measured in Henrys/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the short standard L2 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT:L2 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT:L2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L3?

Description: Sets the short standard L3 inductance coefficient on the selected port of the indicated channel. The L3 coefficient is measured in Henrys/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the short standard L3 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT:L3 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT:L3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:LABEL?

Description: Sets the short standard label on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the short standard label on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT1:SHORT:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:OFFS?

Description: Sets the short standard offset on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the short standard offset on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT:OFFS 2.5E1`

`:SENS1:CORR:COLL:PORT1:SHORT:OFFS?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:SERial?

Description: Sets the short standard serial number on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the short standard serial number on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT:SER '123456'`

`:SENS1:CORR:COLL:PORT1:SHORT:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1

Description: Initiates collection of the Offset Short1 (Short one) standard data for the calibration on the given channel and port. See “Calibration Component Parameters” on page 2-38. for a complete listing of calibration components, connectors, and their Command Parameters. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L0?

Description: Sets the Short1 standard L0 (L zero) inductance on the selected port of the indicated channel. The L0 parameter is measured in Henrys. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard L0 inductance on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1:L0 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT1:L0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L1?

Description: Sets the Short1 standard L1 (L one) inductance coefficient on the selected port of the indicated channel. The L1 coefficient is measured in Henrys/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard L1 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1:L1 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT1:L1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L2?

Description: Sets the Short1 standard L2 inductance coefficient on the selected port of the indicated channel. The L2 coefficient is measured in Henrys/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard L2 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1:L2 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT1:L2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L3?

Description: Sets the Short1 standard L3 inductance coefficient on the selected port of the indicated channel. The L3 coefficient is measured in Henrys/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard L3 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1:L3 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT1:L3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:LABEL?

Description: Sets the Short1 (Short one) standard label on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard label on the selected port of the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT1:SHORT1:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:OFFS?

Description: Sets the Short1 (Short 1) standard offset on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard offset on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1:OFFS 2.5E1`

`:SENS1:CORR:COLL:PORT1:SHORT1:OFFS?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:SERial?

Description: Sets the Short1 (Short 1) standard serial number on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard serial number on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters. See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT1:SER '123456'`

`:SENS1:CORR:COLL:PORT1:SHORT1:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2

Description: Initiates collection of the Offset Short2 standard data for the calibration on the given channel and port.

The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L0?

Description: Sets the Short2 standard L0 (L zero) inductance on the selected port of the indicated channel. The L0 (zero) coefficient is measured in Henrys. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short2 standard L0 inductance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2:L0 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT2:L0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L1?

Description: Sets the Short2 standard L1 (L one) inductance coefficient on the selected port of the indicated channel. The L1 coefficient is measured in Henrys/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short2 standard L1 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2:L1 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT2:L1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L2?

Description: Sets the Short2 standard L2 inductance coefficient on the selected port of the indicated channel. The L2 coefficient is measured in Henrys/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short2 standard L2 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2:L2 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT2:L2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L3?

Description: Sets the Short2 standard L3 inductance coefficient on the selected port of the indicated channel. The L3 coefficient is measured in Henrys/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short2 standard L3 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2:L3 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT2:L3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:LABEL?

Description: Sets the Short2 standard label on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short2 standard label on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT1:SHORT2:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:OFFS?

Description: Sets the Short2 standard offset on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short2 standard offset on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2:OFFS 2.5E1`

`:SENS1:CORR:COLL:PORT1:SHORT2:OFFS?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:SERial?

Description: Sets the Short2 standard serial number on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short2 standard serial number on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters. See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT2:SER '123456'`

`:SENS1:CORR:COLL:PORT1:SHORT2:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3

Description: Initiates collection of the Offset Short3 standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L0?

Description: Sets the Short3 standard L0 (L zero) inductance on the selected port of the indicated channel. The L0 coefficient is measured in Henrys. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short3 standard L0 inductance on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Query Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3:L0 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT3:L0?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L1 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L1?

Description: Sets the Short3 standard L1 (L one) inductance coefficient on the selected port of the indicated channel. The L1 coefficient is measured in Henrys/Hertz. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short3 standard L1 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3:L1 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT3:L1?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L2 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L2?

Description: Sets the Short3 standard L2 inductance coefficient on the selected port of the indicated channel. The L2 coefficient is measured in Henrys/Hertz^2. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short3 standard L2 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^2.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^2.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3:L2 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT3:L2?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L3 <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L3?

Description: Sets the Short3 standard L3 inductance coefficient on the selected port of the indicated channel. The L3 coefficient is measured in Henrys/Hertz^3. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short3 standard L3 inductance coefficient on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Henrys/Hertz^3.

Query Parameters: <NR3> The output parameter is in Henrys/Hertz^3.

Range: MPND

Default Value: See “Calibration Component Parameters” on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3:L3 2.0E0`

`:SENS1:CORR:COLL:PORT1:SHORT3:L3?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:LABEL <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:LABEL?

Description: Sets the Short3 standard label on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard label on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3:LABEL 'IC7000'`

`:SENS1:CORR:COLL:PORT1:SHORT3:LABEL?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:OFFS <NRf>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:OFFS?

Description: Sets the Short3 standard offset on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short3 standard offset on the selected port of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3:OFFS 2.5E1`

`:SENS1:CORR:COLL:PORT1:SHORT3:OFFS?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:SERial <string>
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:SERial?

Description: Sets the Short3 standard serial number on the selected port of the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the Short1 standard serial number on the selected port of the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters. See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SHORT3:SER '123456'`

`:SENS1:CORR:COLL:PORT1:SHORT3:SER?`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD1

Description: Initiates collection of the Sliding Load1 standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SLOAD1`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD2

Description: Initiates collection of the Sliding Load2 standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SLOAD2`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD3

Description: Initiates collection of the Sliding Load3 standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SLOAD3`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD4

Description: Initiates collection of the Sliding Load4 standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SLOAD4`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD5

Description: Initiates collection of the Sliding Load5 standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SLOAD5`

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD6

Description: Initiates collection of the Sliding Load6 standard data for the calibration on the given channel and port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:SLOAD6`

5-78 :SENSe{1-16}:CORRection:COLLect Subsystem

The :SENSe{1-16}:CORRection:COLLect subsystem sets various coefficients and parameters for a pending calibration of the indicated channel.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:COLLect:REFerence:Z0 <NRf>
:SENSe{1-16}:CORRection:COLLect:REFerence:Z0?

Description: Sets the Z0 (Z zero) reference impedance to use for the calibration on the given channel. Outputs the reference impedance to use for the calibration on the given channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: 1E-4 to 1E10

Default Value: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:REF:Z0 7.5E1`

`:SENS1:CORR:COLL:REF:Z0?`

:SENSe{1-16}:CORRection:COLLect:SAVe

Description: Initiates calculation of correction coefficients for the programmed calibration type on the given channel. This command should only be initiated if all required measurements have been collected. If the command is issued before all measurements are completed, an error is generated. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:SAV`

:SENSe{1-16}:CORRection:COLLect:SMCorrection[:STATe] <char>
:SENSe{1-16}:CORRection:COLLect:SMCorrection[:STATe]?

Description: The command sets whether the secondary match correction for calibration will be used.

The query outputs whether the secondary match correction for calibration will be used.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:CORR:COLL:SMC ON`

`:SENS1:CORR:COLL:SMC?`

:SENSe{1-16}:CORRection:COLLect:TFR:CLEar

Description: Clears the list of Transmission Frequency Response calibrations. No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:TFR:CLE`

:SENSe{1-16}:CORRection:COLLect:THRu:ADD <char>

Description: Adds Transmission Throughs (Thrus) to the selected 3-Port Calibration or 4-Port Calibration on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query. The available throughs (thrus) are:

- THRu12 = Through line between Port 1 and Port 2; short form is THR12.
- THRu13 = Through line between Port 1 and Port 3; short form is THR13.
- THRu14 = Through line between Port 1 and Port 4; short form is THR14.
- THRu23 = Through line between Port 2 and Port 3; short form is THR23.
- THRu24 = Through line between Port 2 and Port 4; short form is THR24.
- THRu34 = Through line between Port 3 and Port 4; short form is THR34.

Cmd Parameters: <char> THRu12 | THRu13 | THRu14 | THRu23 | THRu24 | THRu34

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:THR:ADD THRU12`

:SENSe{1-16}:CORRection:COLLect:THRu:CLEar

Description: Clears the Transmission Thrus of the selected Three-Port Calibration or Four-Port Calibration on the indicated channel.

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:THR:CLE`

5-79 :SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem

The :SENSe{1-16}:CORRection:COLLect:WAVeguide subsystem sets the calibration parameters and coefficients for waveguide line types. Use this subsystem to set waveguide parameters before starting a calibration.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:COLLect:WAVeguide:DIELectric <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:DIELectric?

Description: Sets the waveguide calibration kit dielectric value on the indicated channel. Outputs the waveguide calibration kit dielectric value on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: MPND

Default Value: 1.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:DIEL 1.2E0`

`:SENS1:CORR:COLL:WAV:DIEL?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:FREQuency <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:FREQuency?

Description: Sets the waveguide calibration kit cutoff frequency on the indicated channel. Outputs the waveguide calibration kit cutoff frequency on the indicated channel.

Cmd Parameters: <NRf> The input parameter is Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:WAV:FREQ 5.0E9`

`:SENS1:CORR:COLL:WAV:FREQ?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:KIT <char>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:KIT?

Description: Sets the waveguide kit type on the indicated channel. Outputs the waveguide kit type on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <char> WR10 | WR12 | WR15 | USER1 | USER2 | USER3 | USER4 | USER5 | USER6 | USER7 | USER8 | USER9 | USER10 | USER11 | USER12 | USER13 | USER14 | USER15 | USER16 | USER17 | USER18 | USER19 | USER20 | USER21 | USER22 | USER23 | USER24 | USER25 | USER26 | USER27 | USER28 | USER29 | USER30 | USER31 | USER32

Where:

- WR10 = Typical WR-10 RF waveguide
- WR12 = Typical WR-12 RF waveguide
- WR15 = Typical WR-13 RF waveguide
- USERx = User defined waveguide x. The device can be renamed by the user but programmatically is always referred to as “USERx”:

Where x is any number between 1 and 32.

- USER1 = User defined waveguide 1. The device can be renamed by the user but programmatically is always referred to as “USERx”:

Query Parameters: <char> WR10 | WR12 | WR15 | USER1 | USER2 | USER3 | USER4 | USER5 | USER6 | USER7 | USER8 | USER9 | USER10 | USER11 | USER12 | USER13 | USER14 | USER15 | USER16 | USER17 | USER18 | USER19 | USER20 | USER21 | USER22 | USER23 | USER24 | USER25 | USER26 | USER27 | USER28 | USER29 | USER30 | USER31 | USER32

Range: NA

Default Value: USER 1

Syntax Example: `:SENS1:CORR:COLL:WAV:KIT WR12`

`:SENS1:CORR:COLL:WAV:KIT?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:LABel <string>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LABel?

Description: Sets the waveguide calibration kit label on the indicated channel. Outputs the waveguide calibration kit label on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters.

See definition of “<string>” on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: WR10

Syntax Example: `:SENS1:CORR:COLL:WAV:LAB 'wave1'`

`:SENS1:CORR:COLL:WAV:LAB?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:L0 <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:L0?

Description: Sets the waveguide calibration kit L0 (L zero) load inductance on the indicated channel. Outputs the waveguide calibration kit load inductance on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Henrys.

Cmd Parameters: <NR3> The output parameter is in Henrys.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:LOAD:L0 7.5E1`

`:SENS1:CORR:COLL:WAV:LOAD:L0?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:OFFSet <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:OFFSet?

Description: Sets the waveguide calibration kit load offset on the indicated channel. Outputs the waveguide calibration kit load offset on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:LOAD:OFFS 2.5E1`

`:SENS1:CORR:COLL:WAV:LOAD:OFFS?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:R <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:R?

Description: Sets the waveguide calibration kit load resistance on the indicated channel. Outputs the waveguide calibration kit load resistance on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: 5.00000000000E+001

Syntax Example: `:SENS1:CORR:COLL:WAV:LOAD:R 7.5E1`

`:SENS1:CORR:COLL:WAV:LOAD:R?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C0 <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C0?

Description: Sets the waveguide calibration kit open capacitance C0 (C zero) value on the indicated channel. The C0 coefficient is measured in Farads. Outputs the waveguide calibration kit open capacitance C0 value on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Farads.

Query Parameters: <NR3> The output parameter is in Farads.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:OPEN:C0 3.0E-12`

`:SENS1:CORR:COLL:WAV:OPEN:C0?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C1 <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C1?

Description: Sets the waveguide calibration kit open capacitance C1 (C one) term on the indicated channel. The C1 coefficient is measured in Farads/Hertz. Outputs the waveguide calibration kit open capacitance1 term on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz.

Query Parameters: <NR3> The output parameter is in Farads/Hertz.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:OPEN:C1 3.0E-12`

`:SENS1:CORR:COLL:WAV:OPEN:C1?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C2 <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C2?

Description: Sets the waveguide calibration kit open capacitance C2 term on the indicated channel. The C2 coefficient is measured in Farads/Hertz^2. Outputs the waveguide calibration kit open capacitance C2 term on the indicated channel. See “Calibration Component Parameters” on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^2.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^2.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:OPEN:C2 3.0E-12`

`:SENS1:CORR:COLL:WAV:OPEN:C2?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C3 <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C3?

Description: Sets the waveguide calibration kit open capacitance C3 term on the indicated channel. The C3 coefficient is measured in Farads/Hertz^3. Outputs the waveguide calibration kit open capacitance C3 term on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Farads/Hertz^3.

Query Parameters: <NR3> The output parameter is in Farads/Hertz^3.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:OPEN:C3 3.0E-12`

`:SENS1:CORR:COLL:WAV:OPEN:C3?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:OFFSet <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:OFFSet?

Description: Sets the waveguide calibration kit open offset on the indicated channel. Outputs the waveguide calibration kit open offset on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:WAV:OPEN:OFFS 1.2E-4`

`:SENS1:CORR:COLL:WAV:OPEN:OFFS?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:SERial <string>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SERial?

Description: Sets the waveguide calibration kit serial number on the indicated channel. Outputs the waveguide calibration kit serial number on the indicated channel.

Cmd Parameters: <string> The input parameter is any combination of numbers and letters. See definition of "<string>" on page 2-12.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:WAV:SER '12012008'`

`:SENS1:CORR:COLL:WAV:SER?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT1:OFFSet <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT1:OFFSet?

Description: Sets the waveguide calibration kit short1 offset on the indicated channel. Outputs the waveguide calibration kit short1 offset on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:WAV:SHORT1:OFFS 1.2E-4`

`:SENS1:CORR:COLL:WAV:SHORT1:OFFS?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT2:OFFSet <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT2:OFFSet?

Description: Sets the waveguide calibration kit short2 offset on the indicated channel. Outputs the waveguide calibration kit label short2 offset on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:WAV:SHORT2:OFFS 5.0E-9`

`:SENS1:CORR:COLL:WAV:SHORT2:OFFS?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT3:OFFSet <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT3:OFFSet?

Description: Sets the waveguide calibration kit short3 offset length on the indicated channel. Outputs the waveguide calibration kit short3 offset length on the indicated channel. See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: See "Calibration Component Parameters" on page 2-38.

Syntax Example: `:SENS1:CORR:COLL:WAV:SHORT3:OFFS 5.0E9`

`:SENS1:CORR:COLL:WAV:SHORT3:OFFS?`

:SENSe{1-16}:CORRection:COLLect:WAVeguide:SLOAD:MINF <NRf>
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SLOAD:MINF?

Description: Sets the waveguide calibration kit sliding load minimum frequency on the indicated channel. Outputs the waveguide calibration kit sliding load minimum frequency on the indicated channel.See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their Command Parameters.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND

Default Value: 2.00000000000E+009

Syntax Example: `:SENS1:CORR:COLL:WAV:SLOAD:MINF 10.0E7`

`:SENS1:CORR:COLL:WAV:SLOAD:MINF?`

5-80 :SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem

These commands set either a 2-Port Calibration for a 2-Port or 4-Port VNA, or the first 2-Port Calibration (called the :CALa calibration) for a 4-Port VNA. In this case, the second 2-Port Calibration (called the :CALB calibration) is specified by this subsystem

- ":SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs" on page 5-279

Calibration Subsystems with Actual Calibration

Related calibration subsystems that perform actual calibrations are:

- ":SENSe{1-16}:ABORtcal Subsystem" on page 5-264
- ":SENSe{1-16}:CORRection:COLLect:CALB Subsystem - 4-Port VNAs" on page 5-279
- ":SENSe{1-16}:CORRection:COLLect:FLEXible Subsystem" on page 5-288
- ":SENSe{1-16}:CORRection:COLLect:FULL4 Subsystem - 4-Port VNAs" on page 5-289
- ":SENSe{1-16}:CORRection:COLLect:LRL[:CALa] Subsystem" on page 5-317
- ":SENSe{1-16}:CORRection:COLLect:PORT Subsystem" on page 5-336
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380

Calibration Abbreviations

The calibration abbreviations and their calibration types are:

- :1P2PF refers to a one-path two-port calibration forward direction
- :1P2PR refers to a one path two port calibration reverse direction
- :FULL1 refers to a full one port calibration
- :FULL2 refers to a full two port calibration
- :FULL3 refers to a full three port calibration
- :FULLB refers to a full one port reflection calibration on both ports
- :FULL4 refers to a full four port calibration
- :RESP1 refers to a one port response calibration
- :RESPB refers to a one port response calibration both ports
- :TFRB refers to a transmission frequency response calibration both directions
- :TFRF refers to a transmission frequency response calibration forward direction
- :TFRR refers to a transmission frequency response calibration reverse direction

Each calibration simulation type command is described in greater detail in the individual command descriptions below.

Most calibration commands of this type do not have a directly related query. To query the state of these commands, use:

```
:SENSe{1-16}:CORRection:COLLect:TYPe?
```

The :SENSe{1-16}:CORRection:COLLect:PORT subsystem commands start an actual instrument calibration.

:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PF

Description: Sets the first calibration (:CALa) to One-Path Two-Port Forward calibration. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. If a second 2-Port Calibration (:CALB) is required, use the following command:

:SENSe{1-16}:CORRection:COLLect:CALB:1P2PF

No query.

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:1P2PF`

:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PR

Description: Sets the first Two-Port calibration to One-Path Two-Port Reverse on the indicated Channel and port pair. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument.

No query

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:1P2PR`

:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2

Description: Sets the first Two-Port calibration to Full Two-Port on the indicated Channel and port pair. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. No query

Cmd Parameters: NA

Range: NA

Default: NA

Syntax Example: `:SENS1:CORR:COLL:PORT1:FULL2`

5-81 :SENSe{1-16}:CORRection:COLLect Subsystem - 2-Port/4-Port VNA

The :SENSe{1-16}:CORRection:COLLect subsystem commands start an actual calibration and are limited to Port1 and/or Port 2. Both 2-Port and 4-Port VNA instruments can use these commands.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

Related Command Subsystems

These commands are similar to those in the :SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem above where any port pair on a 2-Port or 4-Port VNA can be specified.

- :SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem on page 5-380

Calibration Method Names

Related calibration types use the same command structure of:

```
:SENSe{1-16}:CORRection:COLLect:XXXXX
```

where the calibration XXXXX type is one of the following:

- 1P2PF = One-path two-port calibration, forward direction
- 1P2PR = One path two port calibration, reverse direction
- FULL1 = Full one port calibration
- FULL2 = Full two port calibration
- FULLB = Full one port reflection calibration, both ports
- RESP1 = One port response calibration
- RESPB = One port response calibration, both ports
- TFRB = Transmission frequency response calibration, both directions
- TFRF = Transmission frequency response calibration, forward direction
- TFRR = Transmission frequency response calibration, reverse direction

Each calibration type command is described in greater detail in the following sections. Related to the calibration commands are calibration simulation commands in the general form of :SENSe{1-16}:CORRection:COEFFicient:XXXXX that use the same abbreviations above.

To query the state of this calibration command, use:

```
:SENSe{1-16}:CORRection:COLLect:TYPe?
```

To simulate this calibration, use the following command with the appropriate abbreviation above:

`:SENSe{1-16}:CORRection:COEFFicient:XXXXX`

:SENSe{1-16}:CORRection:COLLect:1P2PF

Description: Selects One-Path Two-Port Forward calibration (1P2PF) as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this calibration command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To simulate this calibration, use:

`:SENSe{1-16}:CORRection:COEFFicient:1P2PF`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:1P2PF`

:SENSe{1-16}:CORRection:COLLect:1P2PR

Description: Selects One-Path Two-Port Reverse calibration (1P2PR) as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To simulate this calibration, use:

`:SENSe{1-16}:CORRection:COEFFicient:1P2PR`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:1P2PR`

:SENSe{1-16}:CORRection:COLLect:ECAL:MSGS:LIST?

Description: Outputs a copy of the Autocal messages list.

Cmd Parameters: NA

Query Parameters: <Arbitrary Block Data> Outputs ECAL List of Messages:

0 - AssurancePassed

1 - Update

2 - TrueThru

3 - Adapter

4 - NoModule

5 - NoOrient

6 - NoFile

7 - NoMatch

8 - No12T

9 - NotAllowed

10 - OutOfRange

11 - AssuranceFailed

12 - Aborted

13 - AbortOK

14 - AbortNotOK

15 - ACError

16 - ACFatalError

17 - DoneCalculateCoeff

18 - ACConnectCalB

19 - CharacBad

20 - DisplayMessage

21 - ConnectToPort1

22 - ConnectToPort2

23 - ConnectToPort3

24 - ConnectToPort4

25 - ConnectToPorts12

26 - ConnectToPorts13

27 - ConnectToPorts14

28 - ConnectToPorts23

29 - ConnectToPorts24

30 - ConnectToPorts34

31 - ConnectThrubwPorts12

32 - ConnectThrubwPorts13

33 - ConnectThrubwPorts14

34 - ConnectThrubwPorts23

35 - ConnectThrubwPorts24

36 - ConnectThrubwPorts34

37 - SequentialBegins

Range: NA

Default: NA

Syntax Example: :SENS1:CORR:COLL:ECAL:MSGS:LIST?

:SENSe{1-16}:CORRection:COLLect:ENHMatch:MIXer:USE:TSM[:STATe]
:SENSe{1-16}:CORRection:COLLect:ENHMatch:MIXer:USE:TSM[:STATe]?

Description: Toggles the True Source Match flag in an Enhanced Match Calibration on the given channel. The query returns the on/off state of the True Source Match flag in an Enhanced Match Calibration on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: :SENS1:CORR:COLL:ENHM:MIX:USE:TSM 1

:SENSe{1-16}:CORRection:COLLect:FULL1

Description: Selects full one port reflection calibration as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. Before sending this command, the simulation port must be specified using:

`:SENSe{1-16}:CORRection:COLLect:PORT`

To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To simulate this calibration, use:

`:SENSe{1-16}:CORRection:COEFFicient:FULL1`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:FULL1`

:SENSe{1-16}:CORRection:COLLect:FULL2

Description: Selects full two port calibration as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To simulate this calibration, use:

`:SENSe{1-16}:CORRection:COEFFicient:FULL2`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:FULL2`

:SENSe{1-16}:CORRection:COLLect:FULLB

Description: Select full one port reflection calibration both ports as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To simulate this calibration, use:

`:SENSe{1-16}:CORRection:COEFFicient:FULLB`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:FULLB`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:CAL1:FILename

Description: Sets the Hybrid Enhanced Match Calibration input cal filename on the given channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:CAL1:FIL 'x:\directory\filename.chx'`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:CAL2:FILename

Description: Sets the Hybrid Enhanced Match Calibration output cal filename on the given channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:CAL2:FIL 'x:\directory\filename.chx'`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:REFPlane:EXTension:MODel
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:REFPlane:EXTension:MODel?

Description: Sets the Hybrid Enhanced Match output adapter type on the given channel. The query returns the Hybrid Enhanced Match output adapter type for the given channel.

Cmd Parameters: <char> TLINe | S2P

Query Parameters: <char> TLIN | S2P

Range: NA

Default Value: TLIN

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:REFP:EXT:MOD S2P`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:S2P:FILename

Description: Sets the Hybrid Enhanced Match Calibration S2P filename on the given channel. No query.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.chx' where x:\directory\filename.chx must exist.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:S2P:FIL 'x:\directory\filename.chx'`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:S2P:REVerse[:STATe]
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:S2P:REVerse[:STATe]?

Description: Toggles the on/off the state of reversing the directional sense of the S2P file on the given channel. The query returns the on/off state of reversing the directional sense of the S2P file for the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:S2P:REV 1`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:DIELectric:TYPe
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:DIELectric:TYPe?

Description: Sets the dielectric type of the Hybrid Enhanced Match TLine model on the given channel. The query returns the dielectric type of the Hybrid Enhanced Match TLine model for the given channel.

Cmd Parameters: <char> AIR|MICROporous|OTHER|POLYethylene|TEFLON

Query Parameters: NA

Query Output: <char> AIR|MICRO|OTHER|POLY|TEFLON

Range: NA

Default: AIR

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:TLIN:DIEL:TYP POLY`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:FREQuency
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:FREQuency?

Description: Sets the Hybrid Enhanced Match TLine reference frequency on the given channel. The query returns the Hybrid Enhanced Match TLine reference frequency for the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Hertz.

Range: MPND

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:TLIN:FREQ 1.0E10`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:IMPedance
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:IMPedance?

Description: Sets the Hybrid Enhanced Match TLine impedance on the given channel. The query returns the Hybrid Enhanced Match TLine impedance for the given channel.

Cmd Parameters: <NRf> The input parameter is in Ohms.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Ohms.

Range: MPND

Default Value: 50.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:TLIN:IMP 7.5E1`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LENGth
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LENGth?

Description: Sets the Hybrid Enhanced Match TLine electrical length on the given channel. The query returns the Hybrid Enhanced Match TLine electrical length for the given channel.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Meters.

Range: NA

Default: 0

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:TLIN:LENG 2.5E-1`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LOSS
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LOSS?

Description: Sets the Hybrid Enhanced Match TLine loss on the given channel. The query returns the Hybrid Enhanced Match TLine loss for the given channel.

Cmd Parameters: <NRf> The input parameter is in dB/mm.

Query Parameters: NA

Query Output: <NR3> The output parameter is in dB/mm.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:TLIN:LOSS 3.0E0`

:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:OTHer
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:OTHer?

Description: Sets the Hybrid Enhanced Match TLine dielectric value for the OTHER dielectric type on the given channel. The query returns the Hybrid Enhanced Match TLine dielectric value for the OTHER dielectric type for the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: 1 to 9.99E3

Default Value: 1.00000000000E+000

Syntax Example: `:SENS1:CORR:COLL:HYBR:ENHM:TLIN:OTH 1.0E3`

:SENSe{1-16}:CORRection:COLLect:LINE <char>
:SENSe{1-16}:CORRection:COLLect:LINE?

Description: Select the line type for calibration of the indicated channel.

Outputs the line type for calibration of the indicated channel.

See "Calibration Component Parameters" on page 2-38 for a complete listing of calibration components, connectors, and their command parameters.

Cmd Parameters: <char> COAXial | MICROstrip | NONDISpersive | WAVEguide

Query Parameters: <char> COAX | MICRO | NONDIS | WAVE

Range: NA

Default Value: COAX

Syntax Example: `:SENS1:CORR:COLL:LINE COAX`

`:SENS1:CORR:COLL:LINE?`

:SENSe{1-16}:CORRection:COLLect:LOAD <char>
:SENSe{1-16}:CORRection:COLLect:LOAD?

Description: Selects the load type broadband/sliding for calibration on the indicated channel. Outputs the load type selection broadband/sliding for calibration on the indicated channel.

Cmd Parameters: <char> FIXed | SLIDing

Query Parameters: <char> FIX | SLID

Range: NA

Default Value: FIX

Syntax Example: `:SENS1:CORR:COLL:LOAD FIX`

`:SENS1:CORR:COLL:LOAD?`

:SENSe{1-16}:CORRection:COLLect:PORT <char>
:SENSe{1-16}:CORRection:COLLect:PORT?

Description: Sets the calibration port for a RESP1 or FULL1 calibration on the indicated channel. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Queries the calibration ports or port pairs on the indicated channel. The command outputs character data for the port combinations. Several examples of various calibration setups are described in the subsections below.

Reflection Frequency Response Calibration

A Reflection Frequency Response Calibration can have up to four reflection frequency response calibrations in one session.

Transmission Frequency Response Calibration

A Transmission Frequency Response Calibration can have up to six transmission frequency response calibrations picked from these types: TFRF, TFRR and TFRB.

One-Port Calibration

A One-Port Calibration can consist of up to four, FULL1 calibrations in one session.

Two-Port Calibration

A Two-Port Calibration can consist of one or two calibrations picked from the following types:

- FULL2, 1P2PF, and 1P2PR.
- The port pair assignments must be independent.
- The Calibration Methods can be Reflection and Thru measurements, AutoCal and LRL/LRM. PORT12 and PORT34 are excluded from LRL/LRM calibrations.
- Both Reflection and Thru measurements and AutoCal can provide FULL2, 1P2PF, and 1P2PR. LRL/LRM calibrations provides FULL2 only.

Three-Port Calibration

A Three-Port Calibration can be accomplished using one of the following techniques:

- Reflection and Thru measurements or two LRL/LRM calibrations.
- PORT12 and PORT34 are excluded from LRL/LRM calibrations.

Four-Port Calibration

A Four-Port Calibration can be accomplished using one of the following techniques:

- Reflection and Thru measurements.
- Two LRL/LRM calibrations
- Two FULL2 AutoCal calibrations.
- PORT12 and PORT34 are excluded from LRL/LRM calibrations.

Cmd Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4 | PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT123 | PORT124 | PORT134 | PORT234 | PORT1234

Query Parameters: <char> PORT1 | PORT2 | PORT3 | PORT4 | PORT12 | PORT13 | PORT14 | PORT23 | PORT24 | PORT34 | PORT123 | PORT124 | PORT134 | PORT234 | PORT1234

Range: NA

Default Value: PORT12

Syntax Example: `:SENS1:CORR:COLL:PORT PORT2`

`:SENS1:CORR:COLL:PORT?`

:SENSe{1-16}:CORRection:COLLect:RESP1

Description: Selects One-Port Response Calibration as the calibration type on the indicated channel. No query. After the method is set, the calibration must be performed. Before sending this command, the simulation port must be specified using:

:SENSe{1-16}:CORRection:COLLect:PORT

To query the state of this command, use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

To simulate this calibration, use:

:SENSe{1-16}:CORRection:COEFFicient:RESP1

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: :SENS1:CORR:COLL:RESP1

:SENSe{1-16}:CORRection:COLLect:RESPB

Description: Selects one port response calibration on both ports as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this command, use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

To simulate this calibration, use:

:SENSe{1-16}:CORRection:COEFFicient:RESPB

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: :SENS1:CORR:COLL:RESPB

:SENSe{1-16}:CORRection:COLLect:TFRB

Description: Selects transmission frequency response calibration both directions as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this command, use:

:SENSe{1-16}:CORRection:COLLect:TYPe?

To simulate this calibration, use:

:SENSe{1-16}:CORRection:COEFFicient:TFRB

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: :SENS1:CORR:COLL:TFRB

:SENSe{1-16}:CORRection:COLLect:TFRF

Description: Selects transmission frequency response calibration forward direction as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To simulate this calibration, use:

`:SENSe{1-16}:CORRection:COEFFicient:TFRF`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:TFRF`

:SENSe{1-16}:CORRection:COLLect:TFRR

Description: Selects transmission frequency response calibration reverse direction as the calibration type on the indicated channel. After the method is set, the calibration must be performed. No query. To query the state of this command, use:

`:SENSe{1-16}:CORRection:COLLect:TYPe?`

To simulate this calibration, use:

`:SENSe{1-16}:CORRection:COEFFicient:TFRR`

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:CORR:COLL:TFRR`

:SENSe{1-16}:CORRection:COLLect:TYPe?

Description: Query only. Outputs the calibration types of the indicated channel. Several examples of various calibration setups are described below.

Reflection Frequency Response Calibration

A Reflection Frequency Response Calibration can have up to four reflection frequency response calibrations in one session. The following is a return for the calibration type query for a response calibration on ports 2, 3 and 4:

`RESP1,RESP1,RESP1`

Transmission Frequency Response Calibration

A Transmission Frequency Response Calibration can have up to six transmission frequency response calibrations picked from these types: TFRF, TFRR and TFRB. The following is the return for the calibration type query for a mix of transmission response calibrations on all possible port pairs:

`TFRF,TFRR,TFRB,TFRR,TFRF,TFRB`

One-Port Calibration

A One-Port Calibration can consist of up to four, FULL1 calibrations in one session. The following is a return for the calibration type query for a FULL1 calibration on PORT134:

`FULL1,FULL1,FULL1`

Two-Port Calibration

A Two-Port Calibration can consist of one or two calibrations picked from the following types: FULL2, 1P2PF, and 1P2PR. The port pair assignments must be independent. The Calibration Methods can be Reflection and Thru measurements, AutoCal and LRL/LRM. PORT12 and PORT34 are excluded from LRL/LRM calibrations. Both Reflection and Thru measurements and AutoCal can provide FULL2, 1P2PF, and 1P2PR. LRL/LRM calibrations provides FULL2 only. The following is a typical return for the calibration type query:

```
1P2PF,FULL2
```

Three Port Calibration

A Three Port Calibration can be accomplished using either Reflection and Thru measurements or two LRL/LRM calibrations. PORT12 and PORT34 are excluded from LRL/LRM calibrations. The following is a return for the calibration type query for each technique:

```
FULL3 | FULL3
```

Four Port Calibration

A Four Port Calibration can be accomplished using Reflection and Thru measurements, two LRL/LRM calibrations, or two FULL2 AutoCal calibrations. PORT12 and PORT34 are excluded from LRL/LRM calibrations. The following is a typical return for the calibration type query for each technique:

```
FULL4 | FULL4
```

Response Calibration Both and Full Calibration Both

The RESPB and FULLB calibration types return the command arguments shown below:

```
RESPB returns RESP1,RESP1
FULLB returns FULL1,FULL1
```

Set of Query Calibration Ports

Use the following commands to set or query the calibration ports:

```
:SENSe{1-16}:CORRection:COLLect:PORT
:SENSe{1-16}:CORRection:COLLect:PORT?
```

Query Parameters: <char>{, <char2>}{, <char3>, <char4>, <char5>, <char6>}

Where the <char> values are combinations of:

- RESP1
- FULL1
- FULL2
- TFRF
- TFRR
- TFRB
- 1P2PF
- 1P2PR

Range: NA

Default Value: FULL2

Syntax Example: `:SENS1:CORR:COLL:TYP?`

5-82 :SENSe{1-16}:CORRection:DISPlay Subsystem - 2-Port VNA

:SENSe{1-16}:CORRection:DISPlay:RSWeep[:STATe] <char>
:SENSe{1-16}:CORRection:DISPlay:RSWeep[:STATe]?

Description: Turns on/off the state of displaying the data when sweeping reverse sweep only on the given channel. Only applies when full 2-port cal is present.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENSe1:CORR:DISP:RSW 1`

`:SENSe1:CORR:DISP:RSW?`

5-83 :SENSe{1-16}:CORRection:EXTension Subsystem

The :SENSe{1-16}:CORRection:EXTension subsystem commands control the reference plane extension from the test ports.

Time Domain, Group Delay, and Reference Plane Subsystems

Related time domain, group delay, and reference place subsystems are:

- Section 5-27 :CALCulate{1-16}:PROCessing:ORDer Subsystem on page 5-115
- Section 5-29 :CALCulate{1-16}:REFerence Subsystem on page 5-127
- Section 5-27 :CALCulate{1-16}:PROCessing:ORDer Subsystem on page 5-115
- Section 5-34 :CALCulate{1-16}[:SELected]:GCOMpression Subsystem on page 5-149
- Section 5-35 :CALCulate{1-16}[:SELected]:GDELay Subsystem on page 5-152
- Section 5-45 :CALCulate{1-16}[:SELected]:TRANsform:TIME Subsystem on page 5-199
- Section 5-83 :SENSe{1-16}:CORRection:EXTension Subsystem on page 5-395

:SENSe{1-16}:CORRection:EXTension:PORT{1-4} <NRf>
:SENSe{1-16}:CORRection:EXTension:PORT{1-4}?

Description: Sets the reference plane extension for the indicated port. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. Outputs the reference plane extension for the indicated port.

Cmd Parameters: <NRf> The input parameter is in Meters.

Query Parameters: <NR3> The output parameter is in Meters.

Range: MPND

Default Value: 0.00000000000E+000

Syntax Example:
```
:SENS1:CORR:EXT:PORT1 3.0E-3
:SENS1:CORR:EXT:PORT1?
```

5-84 :SENSe{1-16}:CORRection:INTerpolation Subsystem

The :SENSe{1-16}:CORRection:INTerpolation subsystem command controls the interpolation state of RF calibration.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:ISOLation Subsystem" on page 5-397
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:INTerpolation:STATe <char>
:SENSe{1-16}:CORRection:INTerpolation:STATe ?

Description: The command Turns RF interpolation on/off for the indicated Channel.

The Querry outputs the RF interpolation on/off status of the indicated Channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Range: NA

Default Value: 0

Syntax Example: `:SENS1:CORR:INT:STAT ON`

`:SENS1:CORR:INT:STAT?`

5-85 :SENSe{1-16}:CORRection:ISOLation Subsystem

The :SENSe{1-16}:CORRection:ISOLation subsystem command controls the use of the isolation data during calibration.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:ISOLation:STATe <char>
:SENSe{1-16}:CORRection:ISOLation:STATe?

Description: For the indicated channel, toggles the use of the isolation coefficient data on/off during correction. Outputs the on/off status of the use of the isolation coefficients during correction for the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:CORR:ISOL:STAT ON`

`:SENS1:CORR:ISOL:STAT?`

5-86 :SENSe{1-16}:CORRection:STATe Subsystem

The :SENSe{1-16}:CORRection:STATe subsystem commands controls the RF correction for one or more channels.

Calibration Setup Subsystems

These subsystems are used during various phases of calibration configuration setup:

- ":CALCulate{1-16}:IMPedance:TRANsformation Subsystem" on page 5-74
- ":CALCulate{1-16}:REFerence Subsystem" on page 5-127
- ":SENSe{1-16}:CORRection:COLLect:METHod Subsystem" on page 5-329
- ":SENSe{1-16}:CORRection:COLLect:MICrostrip Subsystem" on page 5-330
- ":SENSe{1-16}:CORRection:COLLect:MULTIple Subsystem" on page 5-335
- ":SENSe{1-16}:CORRection:COLLect Subsystem" on page 5-371
- ":SENSe{1-16}:CORRection:COLLect:WAVeguide Subsystem" on page 5-373
- ":SENSe{1-16}:CORRection:COLLect[:CALa]:PORT Subsystem" on page 5-380
- ":SENSe{1-16}:CORRection:INTerpolation Subsystem" on page 5-396
- ":SENSe{1-16}:CORRection:STATe Subsystem" on page 5-398

:SENSe{1-16}:CORRection:STATe <char>
:SENSe{1-16}:CORRection:STATe?

Description: For the indicated channel, toggles RF correction on/off. Outputs the RF correction on/off status of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:CORR:STAT ON`

`:SENS1:CORR:STAT?`

5-87 :SENSe{1-16}:FREQuency Subsystem

The :SENSe{1-16}:FREQuency subsystem commands control the various instrument frequencies on a channel-by-channel basis.

Channel and Sweep Subsystems

Related channel and sweep configuration and control subsystems are:

- Section 5-57 :SENSe{1-16}:AVERage Subsystem on page 5-265
- Section 5-87 :SENSe{1-16}:FREQuency Subsystem on page 5-399
- Section 5-99 :SENSe{1-16}:SPUR Subsystem on page 5-489
- Section 5-100 :SENSe{1-16}:SWEep Subsystem on page 5-490

Frequency Limits

The frequency limits for the :SENSe{1-16}:FREQuency subsystem and other commands are described in detail in Chapter 1 “General Information”, “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22 and the related tables in that section. In general, the frequency default values and limits are affected by two factors:

- Instrument Model Number
- Installed Options (Option 70)

:SENSe{1-16}:FREQuency:CENTer <NRf>
:SENSe{1-16}:FREQuency:CENTer?

Description: Sets the center value of the sweep range of the given channel. Outputs the center value of the sweep range of the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22.

Default Value: See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22.

Syntax Example: `:SENS1:FREQ:CENT 10.0E9`

`:SENS1:FREQ:CENT?`

:SENSe{1-16}:FREQuency:CW <NRf>
:SENSe{1-16}:FREQuency:CW?

Description: Sets the CW frequency of the given channel. Outputs the CW frequency of the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22

Default Value: See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22.

Syntax Example: `:SENS1:FREQ:CW 10.0E9`

`:SENS1:FREQ:CW?`

:SENSe{1-16}:FREQuency:DATA <block>
:SENSe{1-16}:FREQuency:DATA?

Description: Enters a new frequency list for the given channel. Outputs the frequency list for the given channel.

Cmd Parameters: See definition of "<block> or <arbitrary block>" on page 2-12. Frequency unit is Hertz.

Query Parameters: See definition of "<block> or <arbitrary block>" on page 2-12. Frequency unit is Hertz.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:FREQ:DATA <block>`

`:SENS1:FREQ:DATA?`

:SENSe{1-16}:FREQuency:SPAN <NRf>
:SENSe{1-16}:FREQuency:SPAN?

Description: Sets the span value of the sweep range of the given channel. Outputs the span value of the sweep range of the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22

Default Value: See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22.

Syntax Example: `:SENS1:FREQ:SPAN 5.0E9`

`:SENS1:FREQ:SPAN?`

:SENSe{1-16}:FREQuency:STARt <NRf>
:SENSe{1-16}:FREQuency:STARt?

Description: Sets the start value of the sweep range of the given channel. Outputs the start value of the sweep range of the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz, Meters, or Seconds.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22

Default Value: See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22.

Syntax Example: `:SENS1:FREQ:STAR 2.0E9`

`:SENS1:FREQ:STAR?`

:SENSe{1-16}:FREQuency:STOP <NRf>
:SENSe{1-16}:FREQuency:STOP?

Description: Sets the stop value of the sweep range of the given channel. Outputs the stop value of the sweep range of the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz, Meters, or Seconds.

Range: See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22.

Default Value: See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22.

Syntax Example: `:SENS1:FREQ:STOP 20.0E9`

`:SENS1:FREQ:STOP?`

5-88 :SENSe{1-16}:FSEGMent Subsystem

The :SENSe{1-16}:FSEGMent subsystem commands are used to configure the active frequency-based segment.

Limit Line and Segment Subsystems

Related limit line and segment configuration and control subsystems are:

- ":CALCulate{1-16}[:SELected]:LIMit Subsystem" on page 5-153
- ":DISPlay Subsystem" on page 5-223
- ":SENSe{1-16}:FSEGMent Subsystem" on page 5-401.
- ":SENSe{1-16}:FSEGMent{1-50} Subsystem" on page 5-412.
- ":SENSe{1-16}:ISEGMent Subsystem" on page 5-436.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446
- ":SENSe{1-16}:SEGMent Subsystem" on page 5-487

:SENSe{1-16}:FSEGMent:ADD

Description: This command adds a new frequency-sweep segment at the end of the frequency-based segment table. No query. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: NA

Range: The range depends on the segment sequence and location:

- Minimum Segment Range = 2 Hz
- Minimum Segment Points = 2 points
- For first segment, Minimum Segment Frequency = Minimum Instrument Frequency.
- For highest frequency entered, Maximum Segment Frequency = Maximum Instrument Frequency.
- For the first segment entered, the Maximum Segment Range = Maximum Instrument Range = (Maximum Instrument Frequency minus Minimum Instrument Frequency).

Default Value: NA

Syntax Example: `:SENS1:FSEGM:ADD`

:SENSe{1-16}:FSEGMent:AVERage:COUNt <NRf>
:SENSe{1-16}:FSEGMent:AVERage:COUNt?

Description: Sets the sweep averaging count in the last frequency-based segment being defined on the indicated channel. Outputs the sweep averaging count in the last frequency-based segment being defined on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 1024

Default Value: 1

Syntax Example: `:SENS1:FSEGM:AVER:COUN 1.01E2`

`:SENS1:FSEGM:AVER:COUN?`

:SENSe{1-16}:FSEGMent:BWIDth[:RESolution] <NRf>
:SENSe{1-16}:FSEGMent:BWIDth[:RESolution]?

Description: Sets the IF bandwidth in the last frequency-based segment being defined on the indicated channel. Outputs the IF bandwidth in the last frequency-based segment being defined on the indicated channel. The system will automatically select the closest IF bandwidth from the following options:

- 1, 3, 10, 30, 100, 300 Hz
- 1, 3, 10, 30, 100, 300 kHz
- 1 MHz

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 1 to 1E6

Default Value: 1.00000000000E+005

Syntax Example: `:SENS1:FSEGM:BWID 3.0E3`

`:SENS1:FSEGM:BWID?`

:SENSe{1-16}:FSEGMent:CLEar

Description: Clears all currently defined segments from the frequency-based segment table, leaving a default segment. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:FSEGM:CLE`

:SENSe{1-16}:FSEGMent:COUNt?

Description: Query only. Outputs the number of segments in the frequency-based segmented sweep on the indicated channel.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 50

Default Value: 1

Syntax Example: `:SENS1:FSEGM:COUN?`

:SENSe{1-16}:FSEGMent:CWMODe[:STATe] <char>
:SENSe{1-16}:FSEGMent:CWMODe[:STATe]?

Description: On the indicated channel, sets the CW mode on/off state in the last frequency-based segment being defined. If CW mode is set, segment has only 1 point. On the indicated channel, outputs the CW mode on/off state in the last frequency-based segment being defined.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:FSEGM:CWMOD ON`

`:SENS1:FSEGM:CWMOD?`

:SENSe{1-16}:FSEGMent:DATA?

Description: Query only. On the indicated channel, outputs the frequency-based segmented sweep table.

Query Parameters: See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:FSEGM:DATA?`

:SENSe{1-16}:FSEGMent:DISPlay <char>
:SENSe{1-16}:FSEGMent:DISPlay?

Description: On the indicated channel, sets the frequency/index display mode for the frequency-based segmented sweep. On the indicated channel, outputs the frequency/index display mode for the frequency-based segmented sweep.

Cmd Parameters: <char> FREQbase | INDEXbase

Query Parameters: <char> FREQ | INDEX

Range: NA

Default Value: FREQ

Syntax Example: `:SENS1:FSEGM:DISP FREQ`

`:SENS1:FSEGM:DISP?`

:SENSe{1-16}:FSEGMent:DISPlay:AVERaging[:STATe] <char>
:SENSe{1-16}:FSEGMent:DISPlay:AVERaging[:STATe]?

Description: Turns on/off the display of the averaging number in all segment tables of the indicated channel. Outputs the on/off state of the display of the averaging number in all segment tables of the indicated channel

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS:FSEGM:DISP:AVER OFF`

`:SENS:FSEGM:DISP:AVER?`

:SENSe{1-16}:FSEGMent:DISPlay:IFBW[:STATe] <char>
:SENSe{1-16}:FSEGMent:DISPlay:IFBW[:STATe]?

Description: Turns on/off the display of the IF Bandwidth in all segment tables of the indicated channel. Outputs the on/off state of the display of the IF Bandwidth in all segment tables of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS:FSEGM:DISP:IFBW ON`

`:SENS:FSEGM:DISP:IFBW?`

:SENSe{1-16}:FSEGMent:DISPlay:POWer[:STATe] <char>
:SENSe{1-16}:FSEGMent:DISPlay:POWer[:STATe]?

Description: Turns on/off the display of the Port Power in all segment tables of the indicated channel. Outputs the on/off state of the display of the Port Power in all segment tables of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS:FSEGM:DISP:POW ON`

`:SENS:FSEGM:DISP:POW?`

:SENSe{1-16}:FSEGMent:FREQuency:ACTive:STARt?

Description: Query only. On the indicated channel, outputs the start frequency of the first active frequency-based segment. The output result is affected the by segment on/off status. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: The default value depends on installed option:

- With Low Frequency Extension Option (070) = 7.00000000000E+004
- Without Option 70 = 10 MHz

Syntax Example: `:SENS1:FSEGM:FREQ:ACT:STAR?`

:SENSe{1-16}:FSEGMent:FREQuency:ACTive:STOP?

Description: Query only. On the indicated channel, outputs the stop frequency of the last active frequency-based segment. The output result is affected the by segment on/off status. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step Size) to Maximum Instrument Frequency

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:FSEGM:FREQ:ACT:STOP?`

:SENSe{1-16}:FSEGMent:FREQuency:FSTArt <NRf>
:SENSe{1-16}:FSEGMent:FREQuency:FSTArt?

Description: On the indicated channel, sets the start frequency in the last frequency-based segment being defined. On the indicated channel, outputs the Segment Start Frequency in the last frequency-based segment being defined. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:FSEGM:FREQ:FSTA 2.0E9`

`:SENS1:FSEGM:FREQ:FSTA?`

:SENSe{1-16}:FSEGMent:FREQuency:FSTEp <NRf>
:SENSe{1-16}:FSEGMent:FREQuency:FSTEp?

Description: On the indicated channel, sets the Segment Frequency Step Size in the last frequency-based segment being defined. On the indicated channel, outputs the Segment Frequency Step Size in the last frequency-based segment being defined. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Frequency Step Size to (Maximum Instrument Frequency minus Minimum Instrument Frequency)

Default Value: The default value depends on installed option:

- With Low Frequency Extension Option 70 = 7.14280714286E+008
- Without Option 70 = 7.13571428571E+008

Syntax Example: `:SENS1:FSEGM:FREQ:FSTE 10.0E3`

`:SENS1:FSEGM:FREQ:FSTE?`

:SENSe{1-16}:FSEGMent:FREQuency:FSTOp <NRf>
:SENSe{1-16}:FSEGMent:FREQuency:FSTOp?

Description: On the indicated channel, sets the Segment Stop Frequency in the last frequency-based segment being defined. On the indicated channel, outputs the Segment Stop Frequency in the frequency-based segment being defined. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step Size) to Maximum Instrument Frequency

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:FSEGM:FREQ:FSTO 10.0E9`

`:SENS1:FSEGM:FREQ:FSTO?`

:SENSe{1-16}:FSEGMent:FREQuency:STARt <NRf>
:SENSe{1-16}:FSEGMent:FREQuency:STARt?

Description: On the indicated channel, sets the Segment Start Frequency of the frequency-based segmented sweep. On the indicated channel, outputs the Segment Start Frequency of the frequency-based segmented sweep. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:FSEGM:FREQ:STAR 10.0E9`

`:SENS1:FSEGM:FREQ:STAR?`

:SENSe{1-16}:FSEGMent:FREQuency:STOP <NRf>
:SENSe{1-16}:FSEGMent:FREQuency:STOP?

Description: On the indicated channel, sets the Segment Stop Frequency of the frequency-based segmented sweep. On the indicated channel, outputs the Segment Stop Frequency of the frequency-based segmented sweep. Available range is limited by the range of the existing segments. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step Size) to Maximum Instrument Frequency

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:FSEGM:FREQ:STOP 20.0E9`

`:SENS1:FSEGM:FREQ:STOP?`

:SENSe{1-16}:FSEGMent:FREQuency[:CW][:FIXed] <NRf>
:SENSe{1-16}:FSEGMent:FREQuency[:CW][:FIXed]?

Description: On the indicated channel, sets the CW Segment Frequency in the last frequency-based segment being defined on the indicated channel. On the indicated channel, outputs the CW Segment Frequency in the frequency-based segment being defined. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR1> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to Maximum Instrument Frequency

Default Value: 70000

Syntax Example: `:SENS1:FSEGM:FREQ 10.0E6`

`:SENS1:FSEGM:FREQ?`

:SENSe{1-16}:FSEGMent:MAXPoints?

Description: Query only. On the indicated channel, outputs the total number of sweep points in the frequency-based segments. For MS464xB Series VNAs, the Maximum Instrument Points (MIP) is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Query Parameters: <NR1> The output parameter is an integer.

Range: Range depends on if CW mode is set:

- If CW is set, range equals 1 (one) point.
- If in sweep mode or non-CW mode, range is from 2 (two) points to Maximum Instrument Points.

Default Value: 15

Syntax Example: `:SENS1:FSEGM:MAXP?`

:SENSe{1-16}:FSEGMent:MODBB:POWer:PORT{1-4} <NRf>
:SENSe{1-16}:FSEGMent:MODBB:POWer:PORT{1-4}?

Description: The command sets the greater than 54 GHz power level on the frequency-based segment being defined on the indicated port and channel. The use of this command requires a MS4647B VNA equipped with Option 80/81. The query outputs the greater than 54 GHz power level on the frequency-based segment being defined on the indicated port and channel.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: NA

Output: <NR3> The output parameter is in dBm.

Range: –60 to +30 dBm

Default Value: The default value depends on model and installed options:

- MS4647B with Option 51, 61, or 62 = –10 dBm
- MS4647B without Option 51, 61, or 62 = –3 dBm

Syntax Example: `:SENS1:FSEGM:MODBB:POW:PORT1 3.0E0`

`:SENS1:FSEGM:MODBB:POW:PORT1?`

:SENSe{1-16}:FSEGMent:POINt?

Description: Query only. On the indicated channel, outputs the displayed number of sweep points in the frequency-based segments. For MS464xB Series VNAs, the Maximum Instrument Points (MIP) is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Query Parameters: <NR1> The output parameter is an integer.

Range: The range depends on if CW mode is set:

- If CW is set, range equals 1 (one) point.
- If in sweep mode or non-CW mode is set, range is from 2 (two) points to Maximum Instrument Points.

Default Value: 15

Syntax Example: `:SENS1:FSEGM:POIN?`

:SENSe{1-16}:FSEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude] <NRf>
:SENSe{1-16}:FSEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]?

Description: On the indicated port and channel, sets the power level on the last frequency-based segment being defined. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. On the indicated port and channel, outputs the power level in the last frequency-based segment being defined.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: –30 to +30

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Option 51, 61, or 62 = –10 dBm
- MS4645B or MS4647B without Option 51, 61, or 62 = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SENS1:FSEGM:POW:PORT1 3.0E0`

`:SENS1:FSEGM:POW:PORT1?`

:SENSe{1-16}:FSEGMent:SPAntype <char>
:SENSe{1-16}:FSEGMent:SPAntype?

Description: On the indicated channel, sets the Segment Span Type of the last frequency-based segment being defined to the specified span type. On the indicated channel, outputs the Segment Span Type of the last frequency-based segment being defined.

STARTSTOP Selected

If STARTSTOP is selected, each segment is defined by the:

- Segment Start Frequency
- Segment Stop Frequency
- Number of Segment Points

STARTSTEP Selected

If STARTSTEP is selected, each segment is defined by the:

- Start Segment Frequency
- Frequency Step Size
- Number of Segment Points

Cmd Parameters: <char> STARTSTOP | STARTSTEP

Query Parameters: <char> STARTSTOP | STARTSTEP

Range: NA

Default Value: STARTSTOP

Syntax Example: `:SENS1:FSEGM:SPA STARTSTOP`

`:SENS1:FSEGM:SPA?`

:SENSe{1-16}:FSEGMent:SWEep:MAXimize

Description: On the indicated channel, maximizes the frequency range of the frequency-based segmented sweep. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:FSEGM:SWE:MAX`

:SENSe{1-16}:FSEGMent:SWEep:POINt <NR1>
:SENSe{1-16}:FSEGMent:SWEep:POINt?

Description: On the indicated channel, sets the number of Segment Sweep Points in the last frequency-based segment being defined. On the indicated channel, outputs the number of sweep points in the last frequency-based segment being defined. If frequency-sweep is set, the range is from 2 (two) points to the Maximum Instrument Points. If the separation between the segment start and stop frequencies is 1 Hz, the number of points is 2 (two). If CW is set, the number of points is 1 (one). For MS464xB Series VNAs, the Maximum Instrument Points (MIP) is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Cmd Parameters: <NR1> The input parameter is an integer.

Query Parameters: <NR1> The output parameter is an integer.

Range: The range depends on if the CW mode is set:

- If CW is set, range = 1 (one) point.
- If in sweep mode or non-CW mode) range = from 2 (two) points to Maximum Instrument Points.

Default Value: 15 or 1 depending on span type.

Syntax Example: `:SENS1:FSEGM:SWE:POIN 1.01E2`

`:SENS1:FSEGM:SWE:POIN?`

:SENSe{1-16}:FSEGMent[:STATe] <char>
:SENSe{1-16}:FSEGMent[:STATe]?

Description: On the indicated channel, toggles the on/off state of the last frequency-based segment being defined. On the indicated channel, outputs the on/off state of the last frequency-based segment being defined.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:FSEGM ON`

`:SENS1:FSEGM?`

5-89 :SENSe{1-16}:FSEGMent{1-50} Subsystem

The :SENSe{1-16}:FSEGMent{1-50} subsystem commands are used to configure the indicated frequency-based segment.

Limit Line and Segment Subsystems

Related limit line and segment configuration and control subsystems are:

- ":CALCulate{1-16}[:SELected]:LIMit Subsystem" on page 5-153
- ":DISPlay Subsystem" on page 5-223
- ":SENSe{1-16}:FSEGMent Subsystem" on page 5-401.
- ":SENSe{1-16}:FSEGMent{1-50} Subsystem" on page 5-412.
- ":SENSe{1-16}:ISEGMent Subsystem" on page 5-436.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446
- ":SENSe{1-16}:SEGMent Subsystem" on page 5-487

:SENSe{1-16}:FSEGMent{1-50}:AVERage:COUNt <NRf>
:SENSe{1-16}:FSEGMent{1-50}:AVERage:COUNt?

Description: On the indicated channel, sets the Sweep Averaging Count in the indicated frequency-based segment. On the indicated channel, outputs the Sweep Averaging Count in the indicated frequency-based segment.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 1024

Default Value: 1

Syntax Example:
```
:SENS1:FSEGM1:AVER:COUN 3
:SENS1:FSEGM1:AVER:COUN?
```

:SENSe{1-16}:FSEGMent{1-50}:BWIDth[:RESolution] <NRf>
:SENSe{1-16}:FSEGMent{1-50}:BWIDth[:RESolution]?

Description: On the indicated channel, sets the Segment IF Bandwidth in the indicated frequency-based segment. On the indicated channel, outputs the Segment IF Bandwidth in the indicated frequency-based segment. The system will automatically select the closest IF bandwidth from the following options:

- 1, 3, 10, 30, 100, 300 Hz
- 1, 3, 10, 30, 100, 300 kHz
- 1 MHz

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 1 to 1E6

Default Value: 1.00000000000E+005

Syntax Example:
```
:SENS1:FSEGM1:BWID 3.0E3
:SENS1:FSEGM1:BWID?
```

:SENSe{1-16}:FSEGMent{1-50}:CWMODe[:STATe] <char>
:SENSe{1-16}:FSEGMent{1-50}:CWMODe[:STATe]?

Description: On the indicated channel, sets the CW mode on/off state in the indicated frequency-based segment. On the indicated channel, outputs the CW mode on/off state in the indicated frequency-based segment.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:FSEGM1:CWMOD ON`

`:SENS1:FSEGM1:CWMOD?`

:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTArt <NRf>
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTArt?

Description: On the indicated channel, sets the Segment Start Frequency in the indicated frequency-based segment. On the indicated channel, outputs the Segment Start Frequency in the indicated frequency-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Query Parameters: <NRf> The input parameter is in Hertz.

Cmd Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:FSEGM1:FREQ:FSTA 3.0E9`

`:SENS1:FSEGM1:FREQ:FSTA?`

:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTEp <NRf>
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTEp?

Description: On the indicated channel, sets the Segment Step Size in the indicated frequency-based segment. Outputs the frequency step size (Fstep) in the indicated frequency-based segment on the indicated channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Frequency Step Size to (Maximum Instrument Frequency minus Minimum Instrument Frequency)

Default Value: The default value depends on the installed option:

- With Low Frequency Extension Option 70 = 7.14280714286E+008
- Without Option 70 = 7.13571428571E+008

Syntax Example: `:SENS1:FSEGM1:FREQ:FSTE 1.00E4`

`:SENS1:FSEGM1:FREQ:FSTE?`

:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTOp <NRf>
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTOp?

Description: On the indicated channel, sets the Segment Stop Frequency in the indicated frequency-based segment. On the indicated channel, outputs the Segment Stop Frequency in the indicated frequency-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step) to Maximum Instrument Frequency

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:FSEGM1:FREQ:FSTO 10.0E9`

`:SENS1:FSEGM1:FREQ:FSTO?`

:SENSe{1-16}:FSEGMent{1-50}:FREQuency[:CW][:FIXed] <NRf>
:SENSe{1-16}:FSEGMent{1-50}:FREQuency[:CW][:FIXed]?

Description: On the indicated channel, sets the Segment CW Frequency in the indicated frequency-based segment. On the indicated channel, outputs the Segment CW Frequency in the indicated frequency-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to Maximum Instrument Frequency

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:FSEGM1:FREQ 5.0E9`

`:SENS1:FSEGM1:FREQ?`

:SENSe{1-16}:FSEGMent{1-50}:MODBB:POWer:PORT{1-4} <NRf>
:SENSe{1-16}:FSEGMent{1-50}:MODBB:POWer:PORT{1-4}?

Description: The command sets the greater than 54 GHz power level for the indicated frequency-based segment, port, and channel. The use of this command requires a MS4647B VNA equipped with Option 80/81. The query outputs the greater than 54 GHz power level for the indicated frequency-based segment, port, and channel.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: NA

Output: <NR3> The output parameter is in dBm.

Range: –60 to +30 dBm

Default Value: The default value depends on model and installed options:

- MS4647B with Option 51, 61, or 62 = –10 dBm
- MS4647B without Option 51, 61, or 62 = –3 dBm

Syntax Example: `:SENS1:FSEGM1:MODBB:POW:PORT1 3.0E0`

`:SENS1:FSEGM1:MODBB:POW:PORT1?`

:SENSe{1-16}:FSEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude] <NRf>
:SENSe{1-16}:FSEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]?

Description: On the indicated port and channel, sets the power level for the indicated frequency-based segment.

On the indicated port and channel, outputs the power level for the indicated frequency-based segment.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: –30 to +30

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SENS1:FSEGM1:POW:PORT1 3.0`

`:SENS1:FSEGM1:POW:PORT1?`

:SENSe{1-16}:FSEGMent{1-50}:SPAntype <char>
:SENSe{1-16}:FSEGMent{1-50}:SPAntype?

Description: On the indicated channel, sets the Segment Span Type of the indicated frequency-based segment. On the indicated channel, outputs the Segment Span Type of the indicated frequency-based segment.

If STARTSTOP is selected, each segment is defined by the:

- Segment Start Frequency
- Segment Stop Frequency
- Number of Segment Points

If STARTSTEP is selected, each segment is defined by the:

- Start Segment Frequency
- Frequency Step Size
- Number of Segment Points

Cmd Parameters: <char> STARTSTOP | STARTSTEP

Query Parameters: <char> STARTSTOP | STARTSTEP

Range: NA

Default Value: STARTSTOP

Syntax Example: `:SENS1:FSEGM1:SPA STARTSTOP`

`:SENS1:FSEGM1:SPA?`

:SENSe{1-16}:FSEGMent{1-50}:SWEep:POINt <NR1>
:SENSe{1-16}:FSEGMent{1-50}:SWEep:POINt?

Description: On the indicated channel, sets the Number of Segment Sweep Points in the indicated frequency-based segment. On the indicated channel, outputs the number of sweep points in the indicated frequency-based segment the indicated channel. The Maximum Instrument Points (MIP) is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Cmd Parameters: <NR1> The input parameter is an integer.

Query Parameters: <NR1> The output parameter is an integer.

Range: Range depends on if CW mode is set.

- If CW is set, 1 point.
- If frequency-sweep is set, range is from 2 (two) points to Maximum Instrument Points.

Default Value: 15 or 1, depending on CW mode.

Syntax Example: `:SENS1:FSEGM1:SWE:POIN 5.01E2`

`:SENS1:FSEGM1:SWE:POIN?`

:SENSe{1-16}:FSEGMent{1-50}[:STATe] <char>
:SENSe{1-16}:FSEGMent{1-50}[:STATe]?

Description: Turns the indicated frequency-based segment on/off on the indicated channel. Outputs the indicated frequency-based segment on/off state on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:FSEGM1`

`:SENS1:FSEGM1?`

5-90 :SENSe{1-16}:GRANging Subsystem

The :SENSe{1-16}:GRANging subsystem commands set the configuration for gain ranging state.

:SENSe{1-16}:GRANging:AUTO[:STATe] <char>
:SENSe{1-16}:GRANging:AUTO[:STATe]?

Description: The command sets the gain ranging state for the specified channel.

The query outputs the gain ranging state for the specified channel.

Note: If you turn auto gain ranging OFF, it will go to the previous manual gain ranging setting (i.e., OFF | LOW | MED | HIGH)

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS:GRAN:AUTO ON`

`:SENS:GRAN:AUTO?`

:SENSe{1-16}:GRANging:MGSTate <char>
:SENSe{1-16}:GRANging:MGSTate?

Description: The command sets the manual gain ranging state for the specified channel.

The query outputs the manual gain ranging state for the specified channel.

Cmd Parameters: <char> OFF | LOW | MED | HIGH

Query Parameters: NA

Query Output: OFF | LOW | MED | HIGH

Range: NA

Default Value: OFF

Syntax Example: `:SENS:GRAN:MGST MED`

`:SENS:GRAN:MGST?`

5-91 :SENSe{1-16}:HOLD Subsystem

The :SENSe(1-16}:HOLD subsystem command sets the hold function for the indicated channel.

Trigger, Hold, and External Source Subsystems

Related trigger, hold, and external source subsystems are:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419
- ":SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem" on page 5-452
- ":SENSe{1-16}:SOURce{1-4} Subsystem" on page 5-488
- ":SOURce:ALL:EXTernal Subsystem" on page 5-508
- ":SOURce{1-4}:EXTernal Subsystem" on page 5-526
- ":TRIGger[:SEQuence] Subsystem" on page 5-552

:SENSe{1-16}:HOLD:FUNCtion <char>
:SENSe{1-16}:HOLD:FUNCtion?

Description: Sets the hold function for the indicated channel. Outputs the hold status for the indicated channel.

Cmd Parameters: <char> CONTinuous | HOLD | SINGle

Query Parameters: <char> CONT | HOLD | SING

Range: NA

Default Value: NA

Syntax Example: `:SENS1:HOLD:FUNC CONT`

`:SENS1:HOLD:FUNC?`

5-92 :SENSe{1-16}:IMD Subsystem

The :SENSe(1-16):IMD Subsystem commands set configuration parameters for IMD and IMDView and controls normalization calibration for the indicated channel.

IMD Configuration and Setup Validation Steps

The IMD setup commands in this subsystem require a sequence of steps to configure and validate the setup:

1. Before changing any settings in the IMD subsystem, the command **:SENSe{1-16}:IMD:CONFigure:DEFine[:STATe] must** be set to ON.
2. After changing IMD subsystem settings, issue the :**SENSe{1-16}:IMD:CONFigure** command.

   (Note: Define mode automatically turns off after issuing the **:IMD:CONF** command and the IMD mode is turned on.)
3. Issue the command **:SENSe{1-16}:IMD:CONFigure:STATus?** to get an explanation of any coercion that occurred.

Note	Once IMD has been configured and turned on, commands from the ":CALCulate{1-16}[:SELected]:DATA Subsystem" can be issued to read data. Normalization calibration can be triggered using ":SENSe{1-16}:IMD:NCALibration:PERForm". Other related calibration subsystems include: :SENSe{1-16}:RECeiver:CALibration commands (located in the ":SENSe{1-16}:RECEiver Subsystem") :SOURce{1-16}:POWer:PORT{1-7}:CORRection commands (located in the ":SOURce{1-16}:POWer Subsystem")

:SENSe{1-16}:IMD:CONFigure

Description: Configures and validates the IMD setup parameters on the indicated channel.

(Note: The command is only available when IMD Define mode is enabled.)

If one or more errors occur during validation then a numeric error code will be added to the system error queue. In order to determine all error codes associated with the number returned by the command, use the following steps:

1. Find the largest number from the list below that is less than or equal to the number returned by the command. The error definition associated with that number has been triggered.
2. Subtract that number from the number returned by the command.
3. Repeat the process with the new number until you reach 0. All numbers found in this process that correspond to any in the list are the errors associated with the numeric error code.

   For example, if the numeric error code returned is 18, you would find the next lower or equal number from the list (16), then subtract it from 18; in this case, 18–16 = 2. Then from the list find the next lower or equal number (2) and subtract it from the previous result; 2–2 = 0. So in this case, the errors associated with the number returned would be 16 (BadState), and 2 (OutOfRange).

1 = PowerOutOfRange

2 = OutOfRange

4 = PointsExceedMax

8 = PointsLessThanMin

16 = BadState

32 = FreqOutOfRange

64 = Tone1SetupInvalid

128 = Tone2SetupInvalid

256 = SetupNotSupported

512 = ToneSetupNotValidForCondor

1024 = InvalidIFBW

2048 = NoResponsesSelected

4096 = InvalidDelta

8192 = InvalidTonesConfiguration

16384 = InvalidResponse

32768 = Invalid

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:IMD:CONF`

:SENSe{1-16}:IMD:CONFigure:DEFine[:STATe] <char>
:SENSe{1-16}:IMD:CONFigure:DEFine[:STATe]?

Description: The command turns the IMD configuration DEFINE mode On/Off on the indicated channel.

The query outputs the IMD configuration DEFINE mode state (On/Off) on the indicated channel.

Cmd Parameters: <char> ON|OFF|1|0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:IMD:CONF:DEF 1`

`:SENS1:IMD:CONF:DEF?`

:SENSe{1-16}:IMD:CONFigure:STATus?

Description: Outputs the IMD configuration and setup status on the indicated channel.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <<Error description>>

Range: NA

Default Value: “Valid”

Syntax Example::SENS1:IMD:CONF:STAT?

:SENSe{1-16}:IMD:DRVer:PORT?

Description: Outputs the driver port setting for the IMD mode on the given channel.

Cmd Parameters: NA

Query Parameters: NA

Query Output: PORT1

PORT2

Range: NA

Default Value: PORT1

Syntax Example: `:SENS1:IMD:DRV:PORT?`

:SENSe{1-16}:IMD:ENABled[:STATe] <char>
:SENSe{1-16}:IMD:ENABled[:STATe]?

Description: The command turns On/Off IMD on the indicated channel.

The query outputs IMD state (On/Off) on the indicated channel.

Cmd Parameters: <char> ON|OFF|1|0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:IMD:ENAB ON`

`:SENS1:IMD:ENAB?`

:SENSe{1-16}:IMD:GRANging:AUTO[:STATe] <char>
:SENSe{1-16}:IMD:GRANging:AUTO[:STATe]?

Description: The command sets the on/off state of the auto gain ranging for the IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the on/off state of the auto gain ranging for the IMD mode of the given channel.

Cmd Parameters: <char> ON|OFF|1|0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:IMD:GRAN:AUTO OFF`

`:SENS1:IMD:GRAN:AUTO?`

:SENSe{1-16}:IMD:GRANging:MGSTate <char>
:SENSe{1-16}:IMD:GRANging:MGSTate?

Description: The command sets the manual gain ranging for the IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the manual gain ranging for the IMD mode of the given channel.

Cmd Parameters: <char> OFF | LOW | MED | HIGH

Query Parameters: NA

Query Output: <char> OFF | LOW | MED | HIGH

Range: NA

Default Value: OFF

Syntax Example: `:SENS1:IMD:GRAN:MGST MED`

`:SENS1:IMD:GRAN:MGST?`

:SENSe{1-16}:IMD:MODe <char>
:SENSe{1-16}:IMD:MODe?

Description: The command sets the IMD mode on the given channel.

The query outputs the IMD mode on the given channel.

Currently available IMD modes:

SLINear = SpectrumView Linear Frequency Sweep

SFSegm = SpectrumView Frequency-based Segmented Sweep

FLINear = Frequency Mode Linear Frequency Sweep

Cmd Parameters: <char> SLINear | SFSegm | FLINear

Query Parameters: NA

Query Output: <char> SLIN | SFS | FLIN

Range: NA

Default Value: FLIN

Syntax Example: `:SENS1:IMD:MOD SLIN`

`:SENS1:IMD:MOD?`

:SENSe{1-16}:IMD:NCALibration:PERForm

Description: Performs IMD normalization calibration on the indicated channel.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:IMD:NCAL:PERF`

:SENSe{1-16}:IMD:NCALibration[:STATe] <char>
:SENSe{1-16}:IMD:NCALibration[:STATe]?

Description: The command turns On/Off the state of IMD normalization calibration on the given channel.

The query outputs the On/Off state of IMD normalization calibration on the given channel.

Cmd Parameters: <char> ON|OFF|1|0

Query Output: <char> 1 | 0

Query Parameters: NA

Range: NA

Default Value: 0

Syntax Example: `:SENS1:IMD:NCAL:STAT 0`

`:SENS1:IMD:NCAL:STAT?`

:SENSe{1-16}:IMD:PARameter{1-16}:DISPlay <char>
:SENSe{1-16}:IMD:PARameter{1-16}:DISPlay?

Description: The command turns On/Off the trace display for the particular trace on the indicated channel.

The query outputs the trace display status (On/Off) for the particular trace on the indicated channel.

Cmd Parameters: <char> ON|OFF|1|0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:IMD:PAR2:DISP 0`

`:SENS1:IMD:PAR2:DISP?`

:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:ORDer <char>
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:ORDer?

Description: The command sets the response order for the given trace in IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

(Note: Command not valid in IMD SpectrumView mode.)

The query outputs the response order for the given trace in IMD mode of the given channel.

Cmd Parameters: <char> FIRSt | SECond | THIRd | FIFTh | SEVenth | NINTh |

Query Parameters: NA

Query Output: FIRS | SEC | THIR | FIFT | SEV | NINT

Range: NA

Default Value: Trace Dependent. For default value, see Table 5-2.

Syntax Example: `:SENS1:IMD:PAR2:RESP:ORD THIR`

`:SENS1:IMD:PAR2:RESP:ORD?`

Table 5-2. :IMD:PARameter{1-16}:RESPonse Commands - Default Values

Trace #	Type	Order	Side	Ref Plane	Ref Type	Display
1	IPO	THIR	PAVG	OUTP	PAVG	ON
2	TPOW	FIRS	PAVG	OUTP	PAVG	ON
3	TPOW	THIR	FLOW	OUTP	PAVG	ON
4	TPOW	THIR	FHIG	OUTP	PAVG	ON
5 to 16	IPO	THIR	PAVG	OUTP	PAVG	OFF

:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:PLANe <char>
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:PLANe?

Description: The command sets the response reference plane for the given trace in IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

(Note: The command not valid in IMD SpectrumView mode.)

The query outputs the response reference plane for the given trace in IMD mode of the given channel.

Cmd Parameters: <char> INPut | OUTPut

Query Parameters: NA

Query Output: INP | OUTP

Range: NA

Default Value: Trace Dependent. For default value, see Table 5-2 on page 5-425.

Syntax Example: `:SENS1:IMD:PAR2:RESP:REF:PLAN OUTP`

`:SENS1:IMD:PAR2:RESP:REF:PLAN?`

:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:TYPE <char>
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:TYPE?

Description: The command sets the response reference type for the given trace in IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

(Note: The command not valid in IMD SpectrumView mode.)

The query outputs the response reference type for the given trace in IMD mode of the given channel.

Currently available Reference Types:

FLOWer = Frequency Lower

FHIGher = Frequency Higher

PAVG = Power Average

PLOWer = Power Lower

PHIGher = Power Higher

Cmd Parameters: <char> FLOWer | FHIGher | PAVG |PLOWer | PHIGher

Query Parameters: NA

Query Output: FLOW | FHIG | PAVG | PLOW | PHIG

Range: NA

Default Value: Trace Dependent. For default value, see Table 5-2 on page 5-425.

Syntax Example: `:SENS1:IMD:PAR2:RESP:REF:TYPE PLOW`

`:SENS1:IMD:PAR2:RESP:REF:TYPE?`

:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:SIDE <char>
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:SIDE?

Description: The command sets the response side for the given trace in IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

(Note: The command not valid in IMD SpectrumView mode.)

The query outputs the response side for the given trace in IMD mode of the given channel.

Currently available Sides:

FLOWer = Frequency Lower

FHIGher = Frequency Higher

PAVG = Power Average

PLOWer = Power Lower

PHIGher = Power Higher

Cmd Parameters: <char> FLOWer | FHIGher | PAVG |PLOWer | PHIGher

Query Parameters: NA

Query Output: FLOW | FHIG | PAVG | PLOW | PHIG

Range: NA

Default Value: Trace Dependent. For default value, see Table 5-2 on page 5-425.

Syntax Example: `:SENS1:IMD:PAR2:RESP:SIDE PLOW`

`:SENS1:IMD:PAR2:RESP:SIDE?`

:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:TYPE <char>
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:TYPE?

Description: The command sets the response Type for the given trace in IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

(Note: When in SpectrumView mode, the type return will be TonePower.)

The query outputs the response Type for the given trace in IMD mode of the given channel.

Currently available Response types:

TPOWer = Tone Power

RTCarrier = Relative To Carrier

IPOint = Intercept Point

TGAin = Tone Gain

ASYMmetry = Asymmetry

Cmd Parameters: <char> TPOWer | RTCarrier | IPOint | TGAin | ASYMmetry

Query Parameters: NA

Query Output: TPOW | RTC | IPO | TGA | ASYM

Range: NA

Default Value: Trace Dependent. For default value, see Table 5-2 on page 5-425.

Syntax Example: `:SENS1:IMD:PAR2:RESP:TYPE TPOW`

`:SENS1:IMD:PAR2:RESP:TYPE?`

:SENSe{1-16}:IMD:POINts <NRf>
:SENSe{1-16}:IMD:POINts?

Description: The command sets the number of points to be used in the IMD measurement on the specified channel.

(Note: The command is only available when IMD Define mode is enabled.)

(Note: This command does not apply when IMD is set to perform a SpectrumView measurement.)

The query outputs the number of points used in the IMD measurement on the specified channel.

Cmd Parameters: <NRf> The input number of points value

Query Parameters: NA

Query Output: <NR1> The output number of points value

Range: 2 to instrument number of points setting (25000)

Default Value: 201

Syntax Example: `:SENS1:IMD:POIN 50`

`:SENS1:IMD:POIN?`

:SENSe{1-16}:IMD:POWer:PORT:COUPle[:STATe] <char>
:SENSe{1-16}:IMD:POWer:PORT:COUPle[:STATe]?

Description: The command turns On/Off the coupled port power of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the On/Off state of the coupled port power of the given channel.

Cmd Parameters: <char> ON | OFF | 1 | 0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENSe{1-16}:IMD:POWer:PORT:COUPle ON`

`:SENSe{1-16}:IMD:POWer:PORT:COUPle?`

:SENSe{1-16}:IMD:PRODuct:IFBW <NRf>
:SENSe{1-16}:IMD:PRODuct:IFBW?

Description: The command sets the product IFBW of IMD mode on the indicated channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the product IFBW of IMD mode on the indicated channel.

Cmd Parameters: <NRf> The input IFBW product value

Query Parameters: NA

Query Output: <NR1> The output IFBW product value

Range: 1 Hz to 1 MHz

Default Value: 1 kHz

Syntax Example: `:SENS1:IMD:PROD:IFBW 10`

`:SENS1:IMD:PROD:IFBW?`

:SENSe{1-16}:IMD:RECeiver:PORT:SELect <char>
:SENSe{1-16}:IMD:RECeiver:PORT:SELect?

Description: The command sets the receiver port of the IMD mode of the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the receiver port of the IMD mode of the given channel.

Cmd Parameters: <char1 > PORT1 | PORT2

Query Parameters: NA

Query Output: PORT1

PORT2

Range: NA

Default Value: PORT2

Syntax Example: `:SENS1:IMD:REC:PORT:SEL PORT1`

`:SENS1:IMD:REC:PORT:SEL?`

:SENSe{1-16}:IMD:SPECtrum:DELTa <NRf>
:SENSe{1-16}:IMD:SPECtrum:DELTa?

Description: The command sets the delta of SpectrumView IMD mode of the given channel.

The query outputs the delta of SpectrumView IMD mode of the given channel.

Cmd Parameters: <NRf> The input delta value of SpectrumView IMD mode

Query Parameters: NA

Query Output: <NR3> The output delta value of SpectrumView IMD mode

Range: Varies based on instrument models and options

Default Value: 3 MHz

Syntax Example: `:SENS1:IMD:SPEC:DELT 10`

`:SENS1:IMD:SPEC:DELT?`

:SENSe{1-16}:IMD:SPECtrum:FREQuency <NRf>
:SENSe{1-16}:IMD:SPECtrum:FREQuency?

Description: The command sets the center frequency of SpectrumView IMD mode of the given channel.

The query outputs the center frequency of SpectrumView IMD mode of the given channel.

Cmd Parameters: <NRf> The input center frequency value

Query Parameters: NA

Query Output: <NR3> The output center frequency value

Range: Varies based on instrument models and options

Default Value: 1 GHz

Syntax Example: `:SENS1:IMD:SPEC:FREQ 1.0E9`

`:SENS1:IMD:SPEC:FREQ?`

:SENSe{1-16}:IMD:SPECtrum:ORDer <char>
:SENSe{1-16}:IMD:SPECtrum:ORDer?

Description: The command sets the highest order of SpectrumView (linear frequency) IMD mode on the indicated channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the highest order of SpectrumView (linear frequency) IMD mode on the indicated channel.

Cmd Parameters: <char> FIRSt | SECond | THIRd | FIFTh | SEVenth | NINTh

Query Parameters: NA

Query Output: FIRS | SEC | THIR | FIFT | SEV | NIN

Range: NA

Default Value: NIN

Syntax Example: `:SENS1:IMD:SPEC:ORD THIR`

`:SENS1:IMD:SPEC:ORD?`

:SENSe{1-16}:IMD:SPECtrum:SEGMented:SECond:ORDer[:STATe] <char>
:SENSe{1-16}:IMD:SPECtrum:SEGMented:SECond:ORDer[:STATe]?

Description: The command turns On/Off the option to include second order of SpectrumView-segmented IMD mode on the indicated channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the second order setting (On/Off) of SpectrumView - segmented IMD mode on the indicated channel.

Cmd Parameters: <char> ON|OFF|1|0

Query Parameters: NA

Range: NA

Default Value: 0

Syntax Example: `:SENS1:IMD:SPEC:SEGM:SEC:ORD:STAT ON`

`:SENS1:IMD:SPEC:SEGM:SEC:ORD:STAT?`

:SENSe{1-16}:IMD:SPECtrum:TONE1:POWer <NRf>
:SENSe{1-16}:IMD:SPECtrum:TONE1:POWer?

Description: The command sets the tone 1 power of SpectrumView IMD mode of the given channel.

The query outputs the tone 1 power of SpectrumView IMD mode of the given channel.

Cmd Parameters: <NRf> The input tone 1 power value

Query Parameters: NA

Query Output: <NR3> The output tone 1 power value

Range: See power range and default for internal sources for your VNA model.

For external synthesizers, default is 0 dBm and range is set by synthesizer limits.

(Note: Max src power is 20 dBm if Option 31 Second Source is installed.)

Default Value: See notes on Range above.

Syntax Example: `:SENS1:IMD:SPEC:TONE1:POW 10`

`:SENS1:IMD:SPEC:TONE1:POW?`

:SENSe{1-16}:IMD:SPECtrum:TONE2:POWer <NRf>
:SENSe{1-16}:IMD:SPECtrum:TONE2:POWer?

Description: The command sets the tone 2 power of SpectrumView IMD mode of the given channel.

The query outputs the tone 2 power of SpectrumView IMD mode of the given channel.

Cmd Parameters: <NRf> The input tone 2 power value

Query Parameters: NA

Query Output: <NR3> The output tone 2 power value

Range: See power range and default for internal sources for your VNA model.

For external synthesizers, default is 0 dBm and range is set by synthesizer limits.

(Note: Max src power is 20 dBm if Option 31 Second Source is installed.)

Default Value: See notes on Range above.

Syntax Example: `:SENS1:IMD:SPEC:TONE2:POW 10`

`:SENS1:IMD:SPEC:TONE2:POW?`

:SENSe{1-16}:IMD:SPUR:AVOidance[:STATe] <char>
:SENSe{1-16}:IMD:SPUR:AVOidance[:STATe]?

Description: The command turns on/off the option to avoid spur on the indicated channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the avoid spur status (On/Off) in IMD configuration on the indicated channel.

Cmd Parameters: <char> ON|OFF|1|0

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:IMD:SPUR:AVO 0`

`:SENS1:IMD:SPUR:AVO?`

:SENSe{1-16}:IMD:SWEPt:DELTa <NRf>
:SENSe{1-16}:IMD:SWEPt:DELTa?

Description: The command sets the delta of swept frequency IMD mode of the given channel.

The query outputs the delta of swept frequency IMD mode of the given channel.

Cmd Parameters: <NRf> The input delta value of swept frequency IMD mode

Query Parameters: NA

Query Output: <NR3> The output delta value of swept frequency IMD mode

Range: Varies based on instrument models and options

Default Value: 3 MHz

Syntax Example: `:SENS1:IMD:SWEP:DELT 10`

`:SENS1:IMD:SWEP:DELT?`

:SENSe{1-16}:IMD:SWEPt:FREQuency:STARt <NRf>
:SENSe{1-16}:IMD:SWEPt:FREQuency:STARt?

Description: The command sets the center start frequency of swept frequency IMD mode of the given channel.

The query outputs the center start frequency of swept frequency IMD mode of the given channel.

Cmd Parameters: <NRf> The center start frequency value.

Query Parameters: NA

Query Output: <NR3> The center start frequency value

Range: Varies based on instrument models and options.

Default Value: Varies based on instrument models and options.

Syntax Example: `:SENS1:IMD:SWEP:FREQ:STAR 1E9`

`:SENS1:IMD:SWEP:FREQ:STAR?`

:SENSe{1-16}:IMD:SWEPt:FREQuency:STOP <NRf>
:SENSe{1-16}:IMD:SWEPt:FREQuency:STOP?

Description: The command sets the center stop frequency of swept frequency IMD mode of the given channel.

The query outputs the center stop frequency of swept frequency IMD mode of the given channel.

Cmd Parameters: <NRf> The center stop frequency value

Query Parameters: NA

Query Output: <NR3> The center stop frequency value

Range: Varies based on instrument models and options

Default Value: Varies based on instrument models and options.

Syntax Example: `SENS1:IMD:SWEP:FREQ:STOP 1E9`

`SENS1:IMD:SWEP:FREQ:STOP?`

:SENSe{1-16}:IMD:SWEPt:TONE1:POWer <NRf>
:SENSe{1-16}:IMD:SWEPt:TONE1:POWer?

Description: The command sets the tone 1 power of swept frequency IMD mode of the given channel.

The query outputs the tone 1 power of swept frequency IMD mode of the given channel.

Cmd Parameters: <NRf> The tone 1 power value

Query Parameters: NA

Query Output: <NR3> The tone 1 power value

Range: See power range and default for internal sources for your VNA model.

For external synthesizers, default is 0 dBm and range is set by synthesizer limits.

(Note: Max src power is 20 dBm if Option 31 Second Source is installed.)

Default Value: See notes on Range above.

Syntax Example: `:SENS1:IMD:SWEP:TONE1:POW 10`

`:SENS1:IMD:SWEP:TONE1:POW?`

:SENSe{1-16}:IMD:SWEPt:TONE2:POWer <NRf>
:SENSe{1-16}:IMD:SWEPt:TONE2:POWer?

Description: The command sets the tone 2 power of swept frequency IMD mode of the given channel.

The query outputs the tone 2 power of swept frequency IMD mode of the given channel.

Cmd Parameters: <NRf> The tone 2 power value

Query Parameters: NA

Query Output: <NR3> The tone 2 power value

Range: See power range and default for internal sources for your VNA model.

For external synthesizers, default is 0 dBm and range is set by synthesizer limits.

(Note: Max src power is 20 dBm if Option 31 Second Source is installed.)

Default Value: See notes on Range above.

Syntax Example: `:SENS1:IMD:SWEP:TONE2:POW 10`

`:SENS1:IMD:SWEP:TONE2:POW?`

:SENSe{1-16}:IMD:TONE1:SOURce <char>
:SENSe{1-16}:IMD:TONE1:SOURce?

Description: The command sets the tone 1 source of the IMD mode on the given channel.

(Note: The command is only available when IMD Define mode is enabled.)

The query outputs the tone 1 source of the IMD mode on the given channel.

Cmd Parameters: Dependent on VNA model and options. See Table 5-3 below.

Query Parameters: NA

Query Output: Dependent on VNA model and options. See Table 5-3 below.

Min-Max Range: NA

Default Value: DT1

Syntax Example: `:SENS1:IMD:TONE1:SOUR S2T1`

`:SENS1:IMD:TONE1:SOUR?`

Table 5-3. Query Output Variables

Query Output Value	Standard No Opt 31	Standard With Opt. 31	3739 Test Set No Opt 31 Freq range >54G	3739 Test Set With Opt. 31 Freq range > 54G	3739 Test Set No Opt 31 Freq range ≤ 54G	3739 Test Set With Opt 31 Freq range ≤ 54G
DT1 (Default To 1)	Int Src ->Port1	Src 1->Port1	Int Src->Mod1	Src 1->Aux Mod	Int Src->Mod 1	Src 1->Mod 1
S2T1 (Source2 To 1)	N/A	Src 2->Port1 (Only applicable if Opt.32 present)	N/A	Src 2->Mod 1	N/A	Src 2->Mod 1 (Only applicable if Opt 32 present)
DT2 (Default To 2)	Int Src ->Port2	Src 2->Port2	Int Src->Mod2	Src 2->Mod 2	Int Src->Mod 2	Src 2->Mod 2
EXT1	Ext Source 1	Ext Source 1	Ext Synth 1 -> Aux Mod	Ext Synth 1 -> Aux Mod	Ext Synth 1	Ext Synth 1
EXT2	Ext Source 2	Ext Source 2	Ext Synth 2 -> Aux Mod	Ext Synth 2 -> Aux Mod	Ext Synth 2	Ext Synth 2
EXT3	Ext Source 3	Ext Source 3	Ext Synth 3 -> Aux Mod	Ext Synth 3 -> Aux Mod	Ext Synth 3	Ext Synth 3
EXT4	Ext Source 4	Ext Source 4	Ext Synth 4 -> Aux Mod	Ext Synth 4 -> Aux Mod	Ext Synth 4	Ext Synth 4

:SENSe{1-16}:IMD:TONE2:SOURce <char>
:SENSe{1-16}:IMD:TONE2:SOURce?

Description: The command sets the tone 2 source of the IMD mode on the given channel.
(Note: The command is only available when IMD Define mode is enabled.)
The query outputs the tone 2 source of the IMD mode on the given channel.

Cmd Parameters: <char> Dependent on VNA model and options. See Table 5-3 on page 5-434.

Query Parameters: NA

Query Output: Dependent on VNA model and options. See Table 5-3 on page 5-434.

Min-Max Range: NA

Default Value: S2T1 (If option 31 and 32 is available)
DT2 (If option 31 is available but option 32 is not available)
EXT1 (If option 31 is not available)

Syntax Example: `:SENS1:IMD:TONE2:SOUR EXT2`
`:SENS1:IMD:TONE2:SOUR?`

:SENSe{1-16}:IMD:TONe:IFBW <NRf>
:SENSe{1-16}:IMD:TONe:IFBW?

Description: The command sets the tone IFBW of IMD mode on the indicated channel.
(Note: The command is only available when IMD Define mode is enabled.)
The query outputs the tone IFBW of IMD mode on the indicated channel.

Cmd Parameters: <NRf> The input IFBW tone value

Query Parameters: NA

Query Output: <NR1> The output IFBW tone value

Range: 1 Hz to 1 MHz

Default Value: 1 kHz

Syntax Example: `:SENS1:IMD:TON:IFBW 10`
`:SENS1:IMD:TON:IFBW?`

5-93 :SENSe{1-16}:ISEGMent Subsystem

The :SENSe{1-16}:ISEGMent subsystem commands are used to configure the active index-based segment.

To configure the index-based segments by segment number, use:

- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446.

Limit Line and Segment Subsystems

Related limit line and segment configuration and control subsystems are:

- ":CALCulate{1-16}[:SELected]:LIMit Subsystem" on page 5-153
- ":DISPlay Subsystem" on page 5-223
- ":SENSe{1-16}:FSEGMent Subsystem" on page 5-401.
- ":SENSe{1-16}:FSEGMent{1-50} Subsystem" on page 5-412.
- ":SENSe{1-16}:ISEGMent Subsystem" on page 5-436.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446
- ":SENSe{1-16}:SEGMent Subsystem" on page 5-487

:SENSe{1-16}:ISEGMent:ADD

Description: For the indicated channel, adds a new segment at the end of the index-based segment table. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:ISEGM:ADD`

:SENSe{1-16}:ISEGMent:AVERage:COUNt <NRf>
:SENSe{1-16}:ISEGMent:AVERage:COUNt?

Description: On the indicated channel, sets the sweep averaging count in the index-based segment being defined. On the indicated channel, outputs the sweep averaging count in the index-based segment being defined. The query channel must be less than or equal to the maximum channel setting.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 1024

Default Value: 1

Syntax Example: `:SENS1:ISEGM:AVER:COUN?`

`:SENS1:ISEGM:AVER:COUN 1.01E2`

:SENSe{1-16}:ISEGMent:BWIDth[:RESolution] <NRf>
:SENSe{1-16}:ISEGMent:BWIDth[:RESolution]?

Description: On the indicated channel, sets the IF bandwidth in the index-based segment being defined. On the indicated channel, outputs the IF bandwidth in the index-based segment being defined. The system will automatically select the closest IF bandwidth from the following options:

- 1, 3, 10, 30, 100, 300 Hz
- 1, 3, 10, 30, 100, 300 kHz
- 1 MHz

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 1 to 1E6

Default Value: 1.00000000000E+005

Syntax Example: `:SENS1:ISEGM:BWID 3.0E4`

`:SENS1:ISEGM:BWID?`

:SENSe{1-16}:ISEGMent:CLEar

Description: On the indicated channel, clears all currently defined segments from the index-based segment table and adds a blank segment. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:ISEGM:CLE`

:SENSe{1-16}:ISEGMent:COUNt?

Description: Query only.

On the indicated channel, outputs the number of segments in the index-based segmented sweep. The query channel must be less than or equal to the maximum channel setting.

Query Parameters: <NR1> The output parameter is an integer.

Range: NA

Default Value: 1

Syntax Example: `:SENS1:ISEGM:COUN?`

:SENSe{1-16}:ISEGMent:CWMODe[:STATe] <char>
:SENSe{1-16}:ISEGMent:CWMODe[:STATe]?

Description: On the indicated channel, sets the CW mode on/off state in the index-based segment being defined. For the indicated channel, returns the CW mode on/off state in the index-based segment being defined.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:ISEGM:CWMOD ON`

`:SENS1:ISEGM:CWMOD?`

:SENSe{1-16}:ISEGMent:DATA?

Description: Query only.

On the indicated channel, outputs the index-based segmented sweep table. The query channel must be less than or equal to the maximum channel setting.

Query Parameters: See definition of “<block> or <arbitrary block>” on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:ISEGM:DATA?`

:SENSe{1-16}:ISEGMent:DISPlay:AVERaging[:STATe] <char>
:SENSe{1-16}:ISEGMent:DISPlay:AVERaging[:STATe]?

Description: Turns on/off the display of the averaging number in all segment tables of the indicated channel. Outputs the on/off state of the display of the averaging number in all segment tables of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS:ISEGM:DISP:AVER ON`

`:SENS:ISEGM:DISP:AVER?`

:SENSe{1-16}:ISEGMent:DISPlay:IFBW[:STATe] <char>
:SENSe{1-16}:ISEGMent:DISPlay:IFBW[:STATe]?

Description: Turns on/off the display of the IF Bandwidth in all segment tables of the indicated channel. Outputs the on/off state of the display of the IF Bandwidth in all segment tables of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: OFF

Syntax Example: `:SENS:ISEGM:DISP:IFBW ON`

`:SENS:ISEGM:DISP:IFBW?`

:SENSe{1-16}:ISEGMent:DISPlay:POWer[:STATe] <char>
:SENSe{1-16}:ISEGMent:DISPlay:POWer[:STATe]?

Description: Turns on/off the display of the Port Power in all segment tables of the indicated channel. Outputs the on/off state of the display of the Port Power in all segment tables of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: OFF

Syntax Example: `:SENS:ISEGM:DISP:POW ON`

`:SENS:ISEGM:DISP:POW?`

:SENSe{1-16}:ISEGMent:FREQuency:FSTArt <NRf>
:SENSe{1-16}:ISEGMent:FREQuency:FSTArt?

Description: On the indicated channel, sets the start frequency in the index-based segment being defined. On the indicated channel, outputs the start frequency in the index-based segment being defined. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to Maximum Instrument Frequency. Step Size = 0 Hz to Maximum Instrument Frequency.

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:ISEGM:FREQ:FSTA 3.0E9`

`:SENS1:ISEGM:FREQ:FSTA?`

:SENSe{1-16}:ISEGMent:FREQuency:FSTEp <NRf>
:SENSe{1-16}:ISEGMent:FREQuency:FSTEp?

Description: On the indicated channel, sets the frequency step size in the index-based segment being defined. On the indicated channel, outputs the frequency step size in the index-based segment being defined. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 0 Hz to Maximum Instrument Frequency

Default Value: 7.14280714286E+008

Syntax Example: `:SENS1:ISEGM:FREQ:FSTE 1.0E9`

`:SENS1:ISEGM:FREQ:FSTE?`

:SENSe{1-16}:ISEGMent:FREQuency:FSTOp <NRf>
:SENSe{1-16}:ISEGMent:FREQuency:FSTOp?

Description: On the indicated channel, sets the stop frequency in the index-based segment being defined. On the indicated channel, outputs the stop frequency in the index-based segment being defined. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to Maximum Instrument Frequency.

Step Size = 0 Hz to Maximum Instrument Frequency.

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:ISEGM:FREQ:FSTO 9.0E9`

`:SENS1:ISEGM:FREQ:FSTO?`

:SENSe{1-16}:ISEGMent:FREQuency[:CW][:FIXed] <NRf>
:SENSe{1-16}:ISEGMent:FREQuency[:CW][:FIXed]?

Description: On the indicated channel, sets the CW frequency in the indicated index-based segment. On the indicated channel, outputs the CW frequency in the indicated index-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to Maximum Instrument Frequency

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:ISEGM:FREQ 1.00E8`

`:SENS1:ISEGM:FREQ?`

:SENSe{1-16}:ISEGMent:INDex:ACTive:STARt?

Description: Query only. On the indicated channel, outputs the start index of the first active index-based segment. The Maximum Instrument Points (MIP) setting is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Cmd Parameters: <NR1> The output parameter is an integer.

Range: The range depends Maximum Instrument Points setting:

- If 25,000 points is set, the range = 0 to 24,999.
- If 100,000 points is set, the range = 0 to 99,999.

Default Value: 0

Syntax Example: `:SENS1:ISEGM:IND:ACT:STAR?`

:SENSe{1-16}:ISEGMent:INDex:ACTive:STOP?

Description: Query only. On the indicated channel, outputs the stop index of the last active index-based segment. The Maximum Instrument Points (MIP) setting is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Query Parameters: <NR1> The output parameter is an integer.

Range: The range depends Maximum Instrument Points setting:

- If 25,000 points is set, the range = 0 to 24,999.
- If 100,000 points is set, the range = 0 to 99,999.

Default Value: 14

Syntax Example: `:SENS1:ISEGM:IND:ACT:STOP?`

:SENSe{1-16}:ISEGMent:INDex:STARt <NRf>
:SENSe{1-16}:ISEGMent:INDex:STARt?

Description: On the indicated channel, sets the start index of the index-based segmented sweep. On the indicated channel, outputs the start index of the index-based segmented sweep. The Maximum Instrument Points (MIP) setting is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: The range depends Maximum Instrument Points setting:

- If 25,000 points is set, the range = 0 to 24,999
- If 100,000 points is set, the range = 0 to 99,999.

Default Value: 0

Syntax Example: `:SENS1:ISEGM:IND:STAR 0`

`:SENS1:ISEGM:IND:STAR?`

:SENSe{1-16}:ISEGMent:INDex:STOP <NRf>
:SENSe{1-16}:ISEGMent:INDex:STOP?

Description: On the indicated channel, sets the stop index of the index-based segmented sweep. On the indicated channel, outputs the stop index of the index-based segmented sweep. The Maximum Instrument Points (MIP) setting is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 14

Default Value: 14

Syntax Example: `:SENS1:ISEGM:IND:STOP?`

`:SENS1:ISEGM:IND:STOP 1`

:SENSe{1-16}:ISEGMent:MAXPoints?

Description: Query only. On the indicated channel, outputs the total number of sweep points in the index-based segments. The number of ISEGMent total points can range from 1 to the Maximum Instrument Points. The Maximum Instrument Points (MIP) setting is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to Maximum Instrument Points

Default Value: 15

Syntax Example: `:SENS1:ISEGM:MAXP?`

:SENSe{1-16}:ISEGMent:MODBB:POWer:PORT{1-4} <NRf>
:SENSe{1-16}:ISEGMent:MODBB:POWer:PORT{1-4}?

Description: The command sets the greater than 54 GHz power level of the index-based segment being defined on the indicated port and channel. The use of this command requires a MS4647B VNA with Option 80/81. The query outputs the greater than 54 GHz power level of the index based segment being defined on the indicated port and channel.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: NA

Output: <NR3> The output parameter is in dBm.

Range: –60 to +30 dBm

Default Value: The default value depends on model and installed options:

- MS4647B with Option 51, 61, or 62 = –10 dBm
- MS4647B without Option 51, 61, or 62 = –3 dBm

Syntax Example: `:SENS1:ISEGM:MODBB:POW:PORT1 3.0E0`

`:SENS1:ISEGM:MODBB:POW:PORT1?`

:SENSe{1-16}:ISEGMent:POINt?

Description: Query only. On the indicated channel, outputs the displayed number of sweep points in the index-based segments. The Maximum Instrument Points (MIP) setting is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Query Parameters: <NR1> The output parameter is an integer.

Range: NA

Default Value: 15

Syntax Example: `:SENS1:ISEGM:POIN?`

:SENSe{1-16}:ISEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]<NRf>
:SENSe{1-16}:ISEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]?

Description: On the indicated port and channel, sets the power level of the index-based segment being defined. The use of Port 3 and/or Port 4 requires a 4-Port VNA instrument. On the indicated port and channel, outputs the power level of the index-based segment being defined.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: -30 to +30

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- MS4645B or MS4647B without Options 051, 061, or 062 = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SENS1:ISEGM:POW:PORT1 3.0E0`

`:SENS1:ISEGM:POW:PORT1?`

:SENSe{1-16}:ISEGMent:SPAntype <char>
:SENSe{1-16}:ISEGMent:SPAntype?

Description: On the indicated channel, sets the span type of the index-based segment being defined as STARTSTOP or STARTSTEP. On the indicated channel, outputs the span type as STARTSTOP or STARTSTEP of the index-based segment being defined.

STARTSTOP Selected

If STARTSTOP is selected, each segment is defined by the:

- Segment Start Frequency
- Segment Stop Frequency
- Number of Segment Points

STARTSTEP Selected

If STARTSTEP is selected, each segment is defined by the:

- Start Segment Frequency
- Frequency Step Size
- Number of Segment Points

Cmd Parameters: <char> STARTSTOP | STARTSTEP

Query Parameters: <char> STARTSTOP | STARTSTEP

Range: NA

Default Value: STARTSTOP

Syntax Example: `:SENS1:ISEGM:SPA STARTSTOP`

`:SENS1:ISEGM:SPA?`

:SENSe{1-16}:ISEGMent:SWEep:MAXimize

Description: On the indicated channel, maximizes the index range of the index-based segmented sweep. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:ISEGM:SWE:MAX`

:SENSe{1-16}:ISEGMent:SWEep:POINt <NR1>
:SENSe{1-16}:ISEGMent:SWEep:POINt?

Description: On the indicated channel, sets the number of sweep points in the index-based segment being defined. On the indicated channel, outputs the number of sweep points in the index-based segment being defined. The Maximum Instrument Points (MIP) setting is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Cmd Parameters: <NR1> The input parameter is an integer.

Query Parameters: <NR1> The output parameter is an integer.

Range: The range depends Maximum Instrument Points setting:

- If set to 25,000 points, the range = 0 to 24,999
- If set to 100,000 points, the range = 0 to 99,999.

Default Value: 15

Syntax Example: `:SENS1:ISEGM:SWE:POIN 15`

`:SENS1:ISEGM:SWE:POIN?`

:SENSe{1-16}:ISEGMent[:STATe] <char>
:SENSe{1-16}:ISEGMent[:STATe]?

Description: On the indicated channel, toggles the index-based segment being defined on and off. On the indicated channel, outputs the on/off state of the index-based segment being defined.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:ISEGM ON`

`:SENS1:ISEGM?`

5-94 :SENSe{1-16}:ISEGMent{1-50} Subsystem

The :SENSe{1-16}:ISEGMent{1-50} subsystem commands are used to configure the indicated index-based segment. To configure only the active index-based segment, use:

- ":SENSe{1-16}:ISEGMent Subsystem" on page 5-436.

Limit Line and Segment Subsystems

Related limit line and segment configuration and control subsystems are:

- ":CALCulate{1-16}[:SELected]:LIMit Subsystem" on page 5-153
- ":DISPlay Subsystem" on page 5-223
- ":SENSe{1-16}:FSEGMent Subsystem" on page 5-401.
- ":SENSe{1-16}:FSEGMent{1-50} Subsystem" on page 5-412.
- ":SENSe{1-16}:ISEGMent Subsystem" on page 5-436.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446
- ":SENSe{1-16}:SEGMent Subsystem" on page 5-487

:SENSe{1-16}:ISEGMent{1-50}:AVERage:COUNt <NRf>
:SENSe{1-16}:ISEGMent{1-50}:AVERage:COUNt?

Description: On the indicated channel, sets the sweep averaging count in the indicated index-based segment. On the indicated channel, outputs the sweep averaging count in the indicated index-based segment.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 1024

Default Value: 1

Syntax Example: `:SENS1:ISEGM1:AVER:COUN 99E0`

`:SENS1:ISEGM1:AVER:COUN?`

:SENSe{1-16}:ISEGMent{1-50}:BWIDth[:RESolution] <NRf>
:SENSe{1-16}:ISEGMent{1-50}:BWIDth[:RESolution]?

Description: On the indicated channel sets the IF bandwidth in the indicated index-based segment. Outputs the IF bandwidth in the indicated index-based segment. The system will automatically select the closest IF bandwidth from the following options:

- 1, 3, 10, 30, 100, or 300 Hz, or
- 1, 3, 10, 30, 100, or 300 kHz, or
- 1 MHz

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: 1 to 1E6

Default Value: 1.00000000000E+005

Syntax Example: `:SENS1:ISEGM1:BWID 3.0E3`

`:SENS1:ISEGM1:BWID?`

:SENSe{1-16}:ISEGMent{1-50}:CWMODe[:STATe] <char>
:SENSe{1-16}:ISEGMent{1-50}:CWMODe[:STATe]?

Description: On the indicated channel, sets the CW mode on/off state in the indicated index-based segment. Outputs the CW mode on/off state in the indicated index-based segment.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:ISEGM1:CWMOD ON`

`:SENS1:ISEGM1:CWMOD?`

:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTArt <NRf>
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTArt?

Description: The command, on the indicated channel, sets the start frequency in the indicated index-based segment. The query outputs the start frequency in the indicated index-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to Maximum Instrument Frequency

Step Size = 0 Hz to Maximum Instrument Frequency

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:ISEGM1:FREQ:FSTA 5.0E9`

`:SENS1:ISEGM1:FREQ:FSTA?`

:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTEp <NRf>
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTEp?

Description: On the indicated channel, sets the frequency step size in the indicated index-based segment. The query outputs the frequency step size in the indicated index-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Frequency Step Size to (Maximum Instrument Frequency minus Minimum Instrument Frequency)

Default Value: 7.14280714286E+008

Syntax Example: `:SENS1:ISEGM1:FREQ:FSTE 1.00E4`

`:SENS1:ISEGM1:FREQ:FSTE?`

:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTOp <NRf>
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTOp?

Description: On the indicated channel, sets the stop frequency in the indicated index-based segment. Outputs the stop frequency in the indicated index-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Range = Minimum Instrument Frequency to Maximum Instrument Frequency. Step Size = 0 Hz to Maximum Instrument Frequency.

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:ISEGM1:FREQ:FSTO 9.0E9`

`:SENS1:ISEGM1:FREQ:FSTO?`

:SENSe{1-16}:ISEGMent{1-50}:FREQuency[:CW][:FIXed] <NRf>
:SENSe{1-16}:ISEGMent{1-50}:FREQuency[:CW][:FIXed]?

Description: On the indicated channel, sets the start frequency in the indicated index-based segment. On the indicated channel, outputs the start frequency in the indicated index-based segment. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to Maximum Instrument Frequency

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:ISEGM1:FREQ 2.0E9`

`:SENS1:ISEGM1:FREQ?`

:SENSe{1-16}:ISEGMent{1-50}:MODBB:POWer:PORT{1-4} <NRf>
:SENSe{1-16}:ISEGMent{1-50}:MODBB:POWer:PORT{1-4}?

Description: The command sets the greater than 54 GHz power level of the indicated index-based segment being defined on the indicated port and channel. The use of this command requires a MS4647B VNA equipped with Option 80/81. The query outputs the greater than 54 GHz power level of the indicated index-based segment being defined on the indicated port and channel.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: NA

Output: <NR3> The output parameter is in dBm.

Range: –60 to +30 dBm

Default Value: The default value depends on model and installed options:

- MS4647B with Option 51, 61, or 62 = –10 dBm
- MS4647B without Option 51, 61, or 62 = –3 dBm

Syntax Example: `:SENS1:ISEGM1:MODBB:POW:PORT1 3.0E0`

`:SENS1:ISEGM1:MODBB:POW:PORT1?`

:SENSe{1-16}:ISEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude] <NRf>
:SENSe{1-16}:ISEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]?

Description: For the indicated channel and port, sets the power level in dBm for the indicated index-based segment.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: –30 to +30

Default Value: Default value depends on model and installed options:

- MS4645B and MS4647B with Options 051, 061, or 062 = – 10 dBm
- All other MS4645B and MS4647B models = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SENS1:ISEGM1:POW:PORT1 3.3E0`

`:SENS1:ISEGM1:POW:PORT1?`

:SENSe{1-16}:ISEGMent{1-50}:SPAntype <char>
:SENSe{1-16}:ISEGMent{1-50}:SPAntype?

Description: Sets the span type of the indicated index-based segment on the indicated channel. Outputs the span type of the indicated index-based segment on the indicated channel.

STARTSTOP Selected

If STARTSTOP is selected, each segment is defined by the:

- Segment Start Frequency
- Segment Stop Frequency
- Number of Segment Points

STARTSTEP Selected

If STARTSTEP is selected, each segment is defined by the:

- Start Segment Frequency
- Frequency Step Size
- Number of Segment Points

Cmd Parameters: <char> STARTSTOP | STARTSTEP

Query Parameters: <char> STARTSTOP | STARTSTEP

Range: NA

Default Value: STARTSTOP

Syntax Example: `:SENS1:ISEGM1:SPA STARTSTOP`

`:SENS1:ISEGM1:SPA?`

:SENSe{1-16}:ISEGMent{1-50}:SWEep:POINt <NR1>
:SENSe{1-16}:ISEGMent{1-50}:SWEep:POINt?

Description: Sets the number of sweep points in the indicated index-based segment on the indicated channel. Outputs the number of sweep points in the indicated index-based segment on the indicated channel. The Maximum Instrument Points (MIP) is either 25,000 or 100,000 points depending on the current MIP setting. The MIP setting is available from the front panel user interface at:

- MAIN | System | SYSTEM | Setup | SETUP | Max Points Setup | MAX POINTS

If the setting is changed from its current setting, a full instrument reboot is required.

Cmd Parameters: <NR1> The input parameter is an integer.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to maximum number of instrument points depending on CW mode. The maximum number of instrument points is either 25,000 or 100,000 points depending on the maximum points setting.

Default Value: 15

Syntax Example: `:SENS1:ISEGM1:SWE:POIN 1.01E2`

`:SENS1:ISEGM1:SWE:POIN?`

:SENSe{1-16}:ISEGMent{1-50}[:STATe] <char>
:SENSe{1-16}:ISEGMent{1-50}[:STATe]?

Description: Turns the indicated index-based segment on/off on the indicated channel. Outputs the indicated index-based segment on/off state on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:ISEGM1`

`:SENS1:ISEGM1?`

5-95 :SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem

The :SENSe{1-16}:OFFSet and :SENSe{1-16}:OFFSet{1-50} subsystem commands are used to configure the frequency offsets for external sources, internal receivers, and related functions for the currently active device or function. Use the :OFFSet{1-50} commands to configure the frequency offsets for external sources, internal receivers, and related functions for the indicated offset number.

Trigger, Hold, and External Source Subsystems

Related trigger, hold, and external source subsystems are:

- :SENSe:HOLD Subsystem on page 5-261
- :SENSe{1-16}:HOLD Subsystem on page 5-419
- :SENSe{1-16}:SOURce{1-4} Subsystem on page 5-488
- :SOURce:ALL:EXTernal Subsystem on page 5-508
- :SOURce{1-4}:EXTernal Subsystem on page 5-526
- :TRIGger[:SEQuence] Subsystem on page 5-552

General Procedure

:SENSe{1-16}:OFFset Commands

Use the :SENSe{1-16}:OFFSet commands to create and setup multiple source bands. The relative sequence is:

- Configure Band 1
  - When started, Band 1 exists with a starting frequency of the lowest instrument frequency, and a top ending frequency of the highest instrument frequency.
  - Change the Band 1 top frequency to something lower than the instrument maximum frequency.
  - Accept the band defaults of receiver source = ON, source state = ON, and delay state = OFF, or configure as required.
- Create and Configure Band 2
  - Create a new band which automatically becomes Band 2 with a starting frequency 1 Hz higher than the end frequency of Band 1 and an end frequency of the highest instrument frequency.
  - If another band is required, configure the end frequency at least 3 Hz lower than the instrument maximum frequency.
  - Accept the band defaults of receiver source = ON, source state = ON, and delay state = OFF or configure as required.
- Create and Configure Band N
  - Create a new band which automatically becomes the next higher Band with a starting frequency 1 Hz higher than the end frequency of the prior Band and an end frequency of the highest instrument frequency.
  - If another band is required, configure the end frequency at least 3 Hz lower than the instrument maximum frequency.
  - If this is the last band, and a lower end frequency is required, configure the band end frequency.
  - Accept the band defaults of receiver source = ON, source state = ON, and delay state = OFF or configure as required.

:SENSe{1-16}:OFFset{1-50} Commands

Use the :SENSe{1-16}:OFFSet{1-50} commands to create and setup multiple source bands. These indexed :OFFSet commands allow a specified existing band to be configured. The relative command sequence is:

- Modify and configure the default existing Band 1 using :OFFSet{1} commands.
  - When started, Band 1 exists with a starting frequency of the lowest instrument frequency, and a top ending frequency of the highest instrument frequency.
  - Change the Band 1 top frequency to something lower than the instrument maximum frequency.
  - Accept the band defaults of receiver source = ON, source state = ON, and delay state = OFF or configure as required.
- Create and Configure Band 2
  - Create a new band using :OFFSet{2} which becomes Band 2 with a starting frequency 1 Hz higher than the end frequency of Band 1 and an end frequency of the highest instrument frequency.
  - If another band is required, configure the end frequency at least 3 Hz lower than the instrument maximum frequency.
  - Accept the band defaults of receiver source = ON, source state = ON, and delay state = OFF or configure as required.
- Create and Configure Band N
  - Create a new band using the :OFFSet{N} (where N is less than or equal to 50) which automatically becomes the next higher Band with a starting frequency 1 Hz higher than the end frequency of the prior Band and an end frequency of the highest instrument frequency.
  - If another band is required, configure the end frequency at least 3 Hz lower than the instrument maximum frequency.
  - If this is the last band, and a lower end frequency is required, configure the band end frequency.
  - Accept the band defaults of receiver source = ON, source state = ON, and delay state = OFF or configure as required.

:SENSe{1-16}:OFFSet:ADD

Description: Adds a multiple source control band at the next position on the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: N/A

Syntax Example: `:SENS1:OFFS:ADD`

:SENSe{1-16}:OFFSet:BBModule:DELay:TIMe <NRf>
:SENSe{1-16}:OFFSet:BBModule:DELay:TIMe?

Description: The command sets the BBModule band start delay for the current multiple source band on the given channel. The query outputs the BBModule start delay for the current multiple source band on the given channel.

Cmd Parameters: <NRf> The parameter units are in seconds.

Query Parameters: <NR3>

Range: 0 to 10 seconds

Default Value: 0.000000E+000

Syntax Example: `:SENS1:OFFS:BBM:DEL:TIM 0.00314`

`:SENS1:OFFS:BBM:DEL:TIM?`

:SENSe{1-16}:OFFSet:BBModule:DELay[:STATe] <char>
:SENSe{1-16}:OFFSet:BBModule:DELay[:STATe]?

Description: Enables/Disables the BBModule band start delay for the current multiple source band on the given channel. Outputs the Enable/Disable status of the BBModule band start delay for the current multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: :SENS1:OFFS:BBM:DEL ON

:SENS1:OFFS:BBM:DEL?

:SENSe{1-16}:OFFSet:BBModule:LEVeling:TYPe <char>
:SENSe{1-16}:OFFSet:BBModule:LEVeling:TYPe?

Description: The command sets the external BBModule leveling type for the current multiple source band on the given channel where:

- IFLEVeling = For Modular BB, IF leveling provides the advantages of RF leveling described below plus the benefit of greater power control range and a lower minimum leveled power (at least 20 dB lower typically). As one approaches a 100 MHz difference between source and receiver frequencies, the power control range will start to decrease as will power control accuracy. The use of IF leveling is not recommended for source-receiver frequency differences greater than 100 MHz.
- RFLEVeling = For Modular BB, RF leveling provides a leveled RF output power with improved source match, protection against over powering the DUT, and insuring that the DUT is operating in its designated power range.

The query returns the BBModule leveling setting for the current multiple source band on the indicated channel.

Cmd Parameters: <char> IFLEVeling | RFLEVeling

Query Parameters: <char> IFLEV | RFLEV

Range: NA

Default Value: NA

Syntax Example: :SENS1:OFFS:BBM:LEV:TYP RFLEV

:SENS1:OFFS:BBM:LEV:TYP?

:SENSe{1-16}:OFFSet:BBModule:RCVR[:STATe]
:SENSe{1-16}:OFFSet:BBModule:RCVR[:STATe]?

Description: Enables/disables the BB Module band receiver for the current multiple source band on the given channel. Outputs the Enable/Disable status of the BBModule receiver for the current multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: :SENS1:OFFS:BBM:RCVR ON

:SENS1:OFFS:BBM:RCVR?

:SENSe{1-16}:OFFSet:BBModule:SRC[:STATe] <char>
:SENSe{1-16}:OFFSet:BBModule:SRC[:STATe]?

Description: Enables/Disables the BBModule band source for the current multiple source band on the given channel. Outputs the Enable/Disable status of the BBModule source for the current multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS:BBM:SRC ON`

`:SENS1:OFFS:BBM:SRC?`

:SENSe{1-16}:OFFSet:CLEar

Description: Clears all currently defined multiple source control bands on the indicated channel, leaving a default band. No query.

Cmd Parameters: NA

Range: NA

Default Value: N/A

Syntax Example: `:SENS1:OFFS:CLE`

:SENSe{1-16}:OFFSet:COMMon:OFFSet[:STATe] <char>
:SENSe{1-16}:OFFSet:COMMon:OFFSet[:STATe]?

Description: Enables the internal source common offset on/off state for the indicated channel. The query returns the internal source common offset on/off state for the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS:COMM:OFFS ON`

`:SENS1:OFFS:COMM:OFFS?`

:SENSe{1-16}:OFFSet:COUNt?

Description: Query only. Outputs the number of multiple source control bands defined on the indicated channel.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 50

Default Value: 1

Syntax Example: `:SENS1:OFFS:COUN?`

:SENSe{1-16}:OFFSet:EXTModule:CONTrol:TYPe <char>
:SENSe{1-16}:OFFSet:EXTModule:CONTrol:TYPe?

Description: The command sets the external module control type on the indicated channel where:

- OFF = All programmatic control of the external module is turned OFF.
- MMW = Programmatic control of a Millimeter-Wave module is turned ON.
- MODMMW = Programmatic control of a Modular Millimeter-Wave module is turned ON.
- MODBB = Programmatic control of a Modular Broadband module is turned ON.
- MODEB = Programmatic control of a Modular E-Band module is turned ON.
- MODWB = Programmatic control of a Modular W-Band module is turned ON.
- MODBB145 = Programmatic control of a Modular 145 GHz Broadband module is turned ON.
- MODBANDED145 = Programmatic control of a Modular 145 GHz Banded module is turned ON

The query outputs the external module control type on the indicated channel.

Note that in turning the External Module Control on or off, this command:

- Cannot determine if a Test Set is attached.
- If a Test Set is attached, cannot determine the Test Set power state.
- If a Test Set is attached, cannot change the Test Set power state (i.e. the command cannot turn the Test Set on/off).
- If a Test Set is attached, cannot determine the Test Set type.

Cmd Parameters: <char> OFF | MMW | MODMMW | MODBB | MODEB | MODWB| MODBB145 | MODBANDED145

Query Parameters: <char> OFF | MMW | MODMMW | MODBB | MODEB | MODWB | MODBB145 | MODBANDED145

Range: NA

Default Value: OFF

Syntax Example: `:SENS1:OFFS:EXTM:CONT:TYP MODBB`

`:SENS1:OFFS:EXTM:CONT:TYP?`

:SENSe{1-16}:OFFSet:DONe

Description: Done defining multiple source control bands on the indicated channel. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS:DON`

:SENSe{1-16}:OFFSet:EXTernal{1-4}:CW[:STATe] <char>
:SENSe{1-16}:OFFSet:EXTernal{1-4}:CW[:STATe]?

Description: For external sources 1 to 4, sets the CW mode on/off status for the last multiple source control band being defined on the indicated channel. Outputs the CW mode on/off status for the last multiple source control band being defined on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS:EXT1:CW ON`

`:SENS1:OFFS:EXT1:CW?`

:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:DIVisor?

Description: For external sources 1 to 4, sets the frequency divisor for the last multiple source control band being defined on the indicated channel. Outputs the frequency divisor for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = ((Multiplier/Divisor) × (Frequency + Offset Frequency))
- If CW is on: Source = ((Multiplier/Divisor) + Offset Frequency)

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:EXT1:DIV 4E0`

`:SENS1:OFFS:EXT1:DIV?`

:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:MULTiplier <NRf>
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:MULTiplier?

Description: For external sources 1 to 4, sets the frequency multiplier for the last multiple source control band being defined on the indicated channel. For external sources 1 to 4, outputs the frequency multiplier for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:EXT1:MULT 2.0E0`

`:SENS1:OFFS:EXT1:MULT?`

:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:OFFSet?

Description: For external sources 1 to 4, sets the offset frequency for the last multiple source control band being defined on the indicated channel. For external sources 1 to 4, outputs the offset frequency for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND. Limited by the band equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS:EXT1:OFFS 1.0E4`

`:SENS1:OFFS:EXT1:OFFS?`

:SENSe{1-16}:OFFSet:EXTModule[:STATe] <char>
:SENSe{1-16}:OFFSet:EXTModule[:STATe]?

Description: Sets the on/off status of external module control in the current multiple source band on the given channel. Outputs the on/off status of external module control in the current multiple source band on the given channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS:EXTM ON`

`:SENS1:OFFS:EXTM?`

:SENSe{1-16}:OFFSet:HIGHest:FREQuency?

Description: Query only.

Outputs the highest multiple source control frequency on the indicated channel. The query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NR3> The output parameter is in Hertz.

Range: MPND. Limited by the band equation.

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:OFFS:HIGH:FREQ?`

:SENSe{1-16}:OFFSet:INTernal{1-2}:CW[:STATe] <char>
:SENSe{1-16}:OFFSet:INTernal{1-2}:CW[:STATe]?

Description: Sets the indicated Internal Source CW mode on/off status for the last multiple source control band being defined on the indicated channel. Outputs the indicated Internal Source CW mode on/off status for the last multiple source control band being defined on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS:INT1:CW ON`

`:SENS1:OFFS:INT1:CW?`

:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:DIVisor?

Description: Sets the internal source 1 frequency divisor for the last multiple source control band being defined on the indicated channel. Outputs the internal source 1 frequency divisor for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:INT1:DIV 4`

`:SENS1:OFFS:INT1:DIV?`

:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:MULTiplier <NRf>
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:MULTiplier?

Description: Sets the internal source 1 frequency multiplier for the last multiple source control band being defined on the indicated channel. Outputs the Internal source 1 or 2 frequency multiplier for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:INT1:MULT 2`

`:SENS1:OFFS:INT1:MULT?`

:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:OFFSet?

Description: Sets the Internal Source 1 or 2 offset frequency for the LAST multiple source control band being defined on the indicated channel. Outputs the Internal Source 1 or 2 offset frequency for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND. Limited by the band equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS:INT1:OFFS 5.0E4`

`:SENS1:OFFS:INT1:OFFS?`

:SENSe{1-16}:OFFSet:INTernal:PHASe <NRf>
:SENSe{1-16}:OFFSet:INTernal:PHASe?

Description: Sets the phase offset between internal sources. Outputs the phase offset between internal sources.

Cmd Parameters: <NRf> The input parameter is in degrees.

Query Parameters: <NR3> The output parameter is in degrees.

Range: –360 to +360

Default Value: 0

Syntax Example: `:SENS1:OFFS:INT:PHAS 1.5E1`

`:SENS1:OFFS:INT:PHAS?`

:SENSe{1-16}:OFFSet:INTernal{1-2}:STATe
:SENSe{1-16}:OFFSet:INTernal{1-2}:STATe?

Description: Sets the state of the indicated internal source on the indicated channel. Outputs the state of the indicated internal source on the indicated channel.

Cmd Parameters: <char> AUTO|ACTive|INACtive

Query Parameters: <char> AUTO|ACT|INAC

Range: NA

Default Value: AUTO

Syntax Example: `:SENS1:OFFS:INT1:STAT INAC`

`:SENS1:OFFS:INT1:STAT?`

:SENSe{1-16}:OFFSet:LOWest:FREQuency?

Description: Query only. Outputs the lowest multiple source control frequency on the indicated channel. The query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND. Limited by the band equation.

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:OFFS:LOW:FREQ?`

:SENSe{1-16}:OFFSet:MMModule:COMMon:MODe[:STATe] <char>
:SENSe{1-16}:OFFSet:MMModule:COMMon:MODe[:STATe]?

Description: The command enables/disables the MMModule common offset mode for the current multiple source band being defined on the given channel. The query outputs the enable/disable status of the MMModule common offset mode for the current multiple source band being defined on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS:MMM:COMM:MOD ON`

`:SENS1:OFFS:MMM:COMM:MOD?`

:SENSe{1-16}:OFFSet:MMModule:DELay:TIMe <char>
:SENSe{1-16}:OFFSet:MMModule:DELay:TIMe?

Description: The command sets the MMModule band start delay for the current multiple source band being defined on the given channel. The query Outputs the MMModule band start delay for the current multiple source band being defined on the given channel

Cmd Parameters: <NRf>

Query Parameters: NA

Query Output: <NR3>

Range: 0 to 10 seconds

Default Value: 0.000000E+000

Syntax Example: `:SENS1:OFFS:MMM:DEL:TIM <NRf>`

`:SENS1:OFFS:MMM:DEL:TIM?`

:SENSe{1-16}:OFFSet:MMModule:DELay[:STATe] <char>
:SENSe{1-16}:OFFSet:MMModule:DELay[:STATe]?

Description: The command enables/disables the MMModule band start delay for the current multiple source band being defined on the given channel. The query outputs the enable/disable status of the MMModule band start delay for the current multiple source band being defined on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS:MMM:DEL 1`

`:SENS1:OFFS:MMM:DEL?`

:SENSe{1-16}:OFFSet:MMModule:LEVeling:TYPe <char>
:SENSe{1-16}:OFFSet:MMModule:LEVeling:TYPe?

Description: The command sets the external MMModule leveling type for the current multiple source band being defined on the given channel. The query outputs the external MMModule leveling type for the current multiple source band being defined on the given channel.

Cmd Parameters: <char> IFLEVeling | RFLEVeling

Query Parameters: NA

Query Output: <char>

Range: NA

Default Value: RFLEV

Syntax Example: `:SENS1:OFFS:MMM:LEV:TYP IFLEV`

`:SENS1:OFFS:MMM:LEV:TYP?`

:SENSe{1-16}:OFFSet:MMModule:RCVR[:STATe] <char>
:SENSe{1-16}:OFFSet:MMModule:RCVR[:STATe]?

Description: The command enables/disables the MMModule receiver for the current multiple source band being defined on the given channel. The query outputs the enable/disable status of the MMModule receiver for the current multiple source band being defined on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:OFFS:MMM:RCVR 1`

`:SENS1:OFFS:MMM:RCVR?`

:SENSe{1-16}:OFFSet:MMModule:SRC[:STATe] <char>
:SENSe{1-16}:OFFSet:MMModule:SRC[:STATe]?

Description: The command enables/disables the MMModule source for the current multiple source band being defined on the given channel. The query outputs the enable/disable status of the MMModule for the current multiple source band being defined on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:OFFS:MMM:SRC ON`

`:SENS1:OFFS:MMM:SRC?`

:SENSe{1-16}:OFFSet:MMModule:VCO[:FREQuency]:OVERrange <NRf>
:SENSe{1-16}:OFFSet:MMModule:VCO[:FREQuency]:OVERrange?

Description: The command sets the MMModule VCO frequency overrange for the current multiple source band being defined on the given channel. The query outputs the MMModule VCO frequency overrange for the current multiple source band being defined on the given channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Hertz.

Range: NA

Default Value: 2.50000000000E+008

Syntax Example: `:SENS1:OFFS:MMM:VCO:OVER <NRf>`

`:SENS1:OFFS:MMM:VCO:OVER?`

:SENSe{1-16}:OFFSet:PINVersion[:STATe] <char>
:SENSe{1-16}:OFFSet:PINVersion[:STATe]?

Description: Turns on/off phase inversion for multiple source control on the indicated channel. Outputs the on/off status of phase inversion for multiple source control on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS:PINV ON`

`:SENS1:OFFS:PINV?`

:SENSe{1-16}:OFFSet:RCVSource:CW[:STATe] <char>
:SENSe{1-16}:OFFSet:RCVSource:CW[:STATe]?

Description: Sets the receiver source CW mode on/off status for the last multiple source control band being defined on the indicated channel. Outputs the receiver source CW mode on/off status for the last multiple source control band being defined on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS:RCVS:CW ON`

`:SENS1:OFFS:RCVS:CW?`

:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:DIVisor?

Description: Sets the receiver source frequency divisor for the last multiple source control band being defined on the indicated channel. Outputs the receiver source frequency divisor for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:RCVS:DIV 3.0E0`

`:SENS1:OFFS:RCVS:DIV?`

:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:MULTiplier <NRf>
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:MULTiplier?

Description: Sets the receiver source frequency multiplier for the last multiple source control band being defined on the indicated channel. Outputs the receiver source frequency multiplier for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:RCVS:MULT 3.0E0`

`:SENS1:OFFS:RCVS:MULT?`

:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:OFFSet?

Description: Sets the receiver source offset frequency for the last multiple source control band being defined on the indicated channel. Outputs the receiver source offset frequency for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND. Limited by the band equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS:RCVS:OFFS 1.0E9`

`:SENS1:OFFS:RCVS:OFFS?`

:SENSe{1-16}:OFFSet:RECEiver:CW[:STATe] <char>
:SENSe{1-16}:OFFSet:RECEiver:CW[:STATe]?

Description: Sets the receiver CW mode on/off status for the last multiple source control band being defined on the indicated channel. On the indicated channel, outputs the receiver CW mode on/off status for the last multiple source control band being defined.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS:RECE:CW ON`

`:SENS1:OFFS:RECE:CW?`

:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:DIVisor?

Description: Sets the receiver frequency divisor for the last multiple source control band being defined on the indicated channel. Outputs the receiver frequency divisor for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:RECE:DIV 4`

`:SENS1:OFFS:RECE:DIV?`

:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:MULTiplier <NRf?
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:MULTiplier?

Description: Sets the receiver frequency multiplier for the last multiple source control band being defined on the indicated channel. Outputs the receiver frequency multiplier for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS:RECE:MULT 2`

`:SENS1:OFFS:RECE:MULT?`

:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:OFFSet?

Description: Sets the receiver offset frequency for the last multiple source control band being defined on the indicated channel. Outputs the receiver offset frequency for the last multiple source control band being defined on the indicated channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: MPND. Limited by the band equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS:RECE:OFFS 1.0E9`

`:SENS1:OFFS:RECE:OFFS?`

:SENSe{1-16}:OFFSet:SPUR:AVOidance[:STATe] <char>
:SENSe{1-16}:OFFSet:SPUR:AVOidance[:STATe]?

Description: Sets internal source spur avoidance on/off state for the indicated channel. The query returns the internal source spur avoidance on/off state for the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:OFFS:SPUR:AVO ON`

`:SENS1:OFFS:SPUR:AVO?`

:SENSe{1-16}:OFFSet:STARt <NRf>
:SENSe{1-16}:OFFSet:STARt?

Description: Sets the start frequency of the last multiple source control band being defined on the indicated channel. Outputs the start frequency of the last multiple source control band being defined on the indicated channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:OFFS:STAR 7.0E4`

`:SENS1:OFFS:STAR?`

:SENSe{1-16}:OFFSet:STOP <NRf>
:SENSe{1-16}:OFFSet:STOP?

Description: Sets the stop frequency of the last multiple source control band being defined on the indicated channel. Outputs the stop frequency of the last multiple source control band being defined on the indicated channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step Size) to Maximum Instrument Frequency

Default Value: 7.00000000000E+010

Syntax Example: `:SENS1:OFFS:STOP 2.0E9`

`:SENS1:OFFS:STOP 2000000000`

`:SENS1:OFFS:STOP?`

:SENSe{1-16}:OFFSet[:STATe] <char>
:SENSe{1-16}:OFFSet[:STATe]?

Description: Sets the state of the multiple source mode of the indicated channel to On, Off, or Define. Outputs the state of the multiple source mode of the indicated channel. Refer to the "General Procedure" on page 5-452 when defining additional bands.

Cmd Parameters: <char> ON | OFF | DEF

Query Parameters: <char> ON | OFF | DEF

Range: NA

Default Value: OFF

Syntax Example: `:SENS1:OFFS ON`

`:SENS1:OFFS?`

:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay:TIMe <NRf>
:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay:TIMe?

Description: Sets the BBModule band start delay for the indicated multiple source band on the given channel. Outputs the BBModule band start delay for the indicated multiple source band on the given channel.

Cmd Parameters: <NRf>

Query Parameters: <NR3>

Range: 0 to 10 seconds

Default Value: 0.000000E+000

Syntax Example: `:SENS1:OFFS1:BBM:DEL:TIM <NRf>`

`:SENS1:OFFS1:BBM:DEL:TIM?`

:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay[:STATe]?

Description: Enables/Disables the BBModule band start delay for the indicated multiple source band on the given channel. Outputs the Enable/Disable status of the BBModule band start delay for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:BBM:DEL ON`

`:SENS1:OFFS1:BBM:DEL?`

:SENSe{1-16}:OFFSet{1-50}:BBModule:LEVeling:TYPe <char>
:SENSe{1-16}:OFFSet{1-50}:BBModule:LEVeling:TYPe?

Description: The command sets the external BBModule leveling type for the specified band on the given channel where:

- IFLEVeling = For Modular BB, IF leveling provides the advantages of RF leveling described below plus the benefit of greater power control range and a lower minimum leveled power (at least 20 dB lower typically). As one approaches a 100 MHz difference between source and receiver frequencies, the power control range will start to decrease as will power control accuracy. The use of IF leveling is not recommended for source-receiver frequency differences greater than 100 MHz.
- RFLEVeling = For Modular BB, RF leveling provides a leveled RF output power with improved source match, protection against over powering the DUT, and insuring that the DUT is operating in its designated power range.

The query returns the BBModule leveling setting for the current multiple source band on the indicated channel.

Cmd Parameters: <char> IFLEVeling | RFLEVeling

Query Parameters: <char> IFLEV | RFLEV

Range: NA

Default Value: RFLEV

Syntax Example: `:SENS1:OFFS1:BBM:LEV:TYP IFLEV`

`:SENS1:OFFS1:BBM:LEV:TYP?`

:SENSe{1-16}:OFFSet{1-50}:BBModule:RCVR[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:BBModule:RCVR[:STATe]?

Description: Enables/Disables the BBModule band receiver for the indicated multiple source band on the given channel. Outputs the Enable/Disable status of the BBModule receiver for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:OFFS1:BBM:RCVR ON`

`:SENS1:OFFS1:BBM:RCVR?`

:SENSe{1-16}:OFFSet{1-50}:BBModule:SRC[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:BBModule:SRC[:STATe]?

Description: The command enables or disables the BBModule band source for the indicated multiple source band on the given channel. The query outputs the Enable/Disable status of the BBModule source for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:OFFS1:BBM:SRC ON`

`:SENS1:OFFS1:BBM:SRC?`

:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}:CW[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}:CW[:STATe]?

Description: For external sources 1 to 4, sets the CW mode on/off status for the indicated multiple source band and channel. For external sources 1 to 4, outputs the CW mode on/off status for the indicated multiple source band and channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:EXT1:CW ON`

`:SENS1:OFFS1:EXT1:CW?`

:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:DIVisor?

Description: For external sources 1 to 4, sets the frequency divisor for the indicated multiple source band and channel. For external source 1 to 4, outputs the frequency divisor for the indicated multiple source band and channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: Depends on multiple source equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS1:EXT1:DIV 2`

`:SENS1:OFFS1:EXT1:DIV?`

:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:MULTiplier <NRf>
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:MULTiplier?

Description: For external source 1 to 4, sets the frequency multiplier for the indicated multiple source band and channel. For external sources 1 to 4, outputs the frequency multiplier for the indicated multiple source band and channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS1:EXT1:MULT 4`

`:SENS1:OFFS1:EXT1:MULT?`

:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:OFFSet?

Description: Sets the offset frequency for the indicated external source for the indicated multiple source band and channel. The multiple source equations depend on whether the instrument is in Broadband or Millimeter-Wave Mode.

- If the VNA is in Broadband Mode, the Start, Stop, and Broadband Break Frequencies in GHz set the frequency range.
- If the VNA is in Millimeter-Wave Mode, only the Start and Stop Frequency in GHz are required.

The resultant equations for LO and RF are:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Outputs one of the four available external source’s offset frequency for the indicated multiple source band and channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Depends on multiple source equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS1:EXT1:OFFS 1.002`

`:SENS1:OFFS1:EXT1:OFFS?`

:SENSe{1-16}:OFFSet{1-50}:EXTModule[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:EXTModule[:STATe]?

Description: Sets the on/off status of external module control in the given multiple source band on the given channel. Outputs the on/off status of external module control of the given multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:EXTM ON`

`:SENS1:OFFS1:EXTM?`

:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}:CW[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}:CW[:STATe]?

Description: Sets the Internal Source 1 or Internal Source 2 CW mode on/off status for the indicated multiple source band and channel. Outputs the Internal Source 1 or Internal Source 2 CW mode on/off status for the indicated multiple source band and channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:INT1:CW ON`

`:SENS1:OFFS1:INT1:CW?`

:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:DIVisor?

Description: Sets the Internal source 1 frequency divisor for the indicated multiple source band and channel. The multiple source equations depend on whether the instrument is in Broadband or Millimeter-Wave Mode.

- If the VNA is in Broadband Mode, the Start, Stop, and Broadband Break Frequencies in GHz set the frequency range.
- If the VNA is in Millimeter-Wave Mode, only the Start and Stop Frequency in GHz are required.

The resultant equations for LO and RF are:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Outputs the Internal source 1 frequency divisor for the indicated multiple source band and channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: Depends on the multiple source equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS1:INT1:DIV 2`

`:SENS1:OFFS1:INT1:DIV?`

:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:MULTiplier <NRf>
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:MULTiplier?

Description: Sets the Internal source 1 frequency multiplier for the indicated multiple source band and channel. Outputs the Internal source 1 frequency multiplier for the indicated multiple source band and channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS1:INT1:MULT 4`

`:SENS1:OFFS1:INT1:MULT?`

:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:OFFSet?

Description: Sets the Internal source 1 offset frequency for the indicated multiple source band and channel. The multiple source equations depend on whether the instrument is in Broadband or Millimeter-Wave Mode.

- If the VNA is in Broadband Mode, the Start, Stop, and Broadband Break Frequencies in GHz set the frequency range.
- If the VNA is in Millimeter-Wave Mode, only the Start and Stop Frequency in GHz are required.

The resultant equations for LO and RF are:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Outputs the Internal source 1 offset frequency for the indicated multiple source band and channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Depends on the multiple source equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS1:INT1:OFFS 1.0`

`:SENS1:OFFS1:INT1:OFFS?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe]?

Description: The command enables/disables the MMModule common offset mode for the indicated multiple source band on the given channel. The query outputs the enable/disable status of the MMModule common offset mode for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:MMM:COMM:MOD ON`

`:SENS1:OFFS1:MMM:COMM:MOD?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe <NRf>
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe?

Description: The command sets the MMModule band start delay for the indicated multiple source band on the given channel. The query outputs the MMModule band start delay for the indicated multiple source band on the given channel

Cmd Parameters: <NRf>

Query Parameters: NA

Query Output: <NR3>

Range: 0 to 10 seconds

Default Value: 0.000000E+000

Syntax Example: `:SENS1:OFFS1:MMM:DEL:TIM <NRf>`

`:SENS1:OFFS1:MMM:DEL:TIM?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe]?

Description: The command enables/disables the MMModule band start delay for the indicated multiple source band on the given channel. The query outputs the enable/disable status of the MMModule band start delay for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:MMM:DEL ON`

`:SENS1:OFFS1:MMM:DEL?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe <char>
:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe?

Description: The command sets the external MMModule leveling type for the indicated multiple source band on the given channel. The query outputs the external MMModule leveling type for the indicated multiple source band on the given channel.

- IFLEVeling = For Modular BB, IF leveling provides the advantages of RF leveling described below plus the benefit of greater power control range and a lower minimum leveled power (at least 20 dB lower typically). As one approaches a 100 MHz difference between source and receiver frequencies, the power control range will start to decrease as will power control accuracy. The use of IF leveling is not recommended for source-receiver frequency differences greater than 100 MHz.
- RFLEVeling = For Modular BB, RF leveling provides a leveled RF output power with improved source match, protection against over powering the DUT, and insuring that the DUT is operating in its designated power range.

Cmd Parameters: <char> IFLEVeling | RFLEVeling

Query Parameters: NA

Query Output: <char> IFLEV | RFLEV

Range: NA

Default Value: RFLEV

Syntax Example: `:SENS1:OFFS1:MMM:LEV:TYP IFLEV`

`:SENS1:OFFS1:MMM:LEV:TYP?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe]?

Description: The command enables/disables the MMModule receiver for the indicated multiple source band on the given channel. The query outputs the enable/disable status of the MMModule receiver for the indicated multiple source band on the given channel

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SENS1:OFFS1:MMM:RCVR ON`

`:SENS1:OFFS1:MMM:RCVR?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe]?

Description: The command enables/disables the MMModule source for the indicated multiple source band on the given channel. The query outputs the enable/disable status of the MMModule source for the indicated multiple source band on the given channel

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:SRC ON`

`:SENS1:OFFS1:MMM:SRC?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange <NRf>
:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange?

Description: The command sets the MMModule VCO frequency overrange for the indicated multiple source band on the given channel. The query outputs the MMModule VCO frequency overrange for the indicated multiple source band on the given channel

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Hertz.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:VCO:OVER <NRf>`

`:SENS1:OFFS1:MMM:VCO:OVER?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe]?

Description: The command enables/disables the MMModule common offset mode for the indicated multiple source band on the given channel. The query outputs the enable/disable status of the MMModule common offset mode for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:MMM:COMM:MOD 1`

`:SENS1:OFFS1:MMM:COMM:MOD?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe?

Description: The command sets the MMModule band start delay for the indicated multiple source band on the given channel. The query outputs the MMModule band start delay for the indicated multiple source band on the given channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Query Output: <NR3>

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:DEL:TIM <NRf>`

`:SENS1:OFFS1:MMM:DEL:TIM?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe]
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe]?

Description: The command enables/disables the MMModule band start delay for the indicated multiple source band on the given channel. The query outputs the enable/disable status of the MMModule band start delay for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:DEL 1`

`:SENS1:OFFS1:MMM:DEL?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe
:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe?

Description: The command sets the external MMModule leveling type for the indicated multiple source band on the given channel. The query outputs the external MMModule leveling type for the indicated multiple source band on the given channel.

- IFLEVeling = For Modular BB, IF leveling provides the advantages of RF leveling described below plus the benefit of greater power control range and a lower minimum leveled power (at least 20 dB lower typically). As one approaches a 100 MHz difference between source and receiver frequencies, the power control range will start to decrease as will power control accuracy. The use of IF leveling is not recommended for source-receiver frequency differences greater than 100 MHz.
- RFLEVeling = For Modular BB, RF leveling provides a leveled RF output power with improved source match, protection against over powering the DUT, and insuring that the DUT is operating in its designated power range.

Cmd Parameters: <char> IFLEVeling | RFLEVeling

Query Parameters: NA

Query Output: <char> IFLEV | RFLEV

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:LEV:TYP IFLEV`

`:SENS1:OFFS1:MMM:LEV:TYP?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe]
:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe]?

Description: The command enables/disables the MMModule receiver for the indicated multiple source band on the given channel. The query outputs the enable/disable status of the MMModule receiver for the indicated multiple source band on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:RCVR 1`

`:SENS1:OFFS1:MMM:RCVR?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe]
:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe]?

Description: The command enables/disables the MMModule source for the indicated multiple source band on the given channel. The query outputs the Enable/Disable status of the MMModule source for the indicated multiple source band on the given channel

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:SRC 1`

`:SENS1:OFFS1:MMM:SRC?`

:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange
:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange?

Description: The command sets the MMModule VCO frequency overrange for the indicated multiple source band on the given channel. The query outputs the MMModule VCO frequency overrange for the indicated multiple source band on the given channel

Cmd Parameters: <NRf>: The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3>: The output parameter is in Hertz.

Range: NA

Default Value: NA

Syntax Example: `:SENS1:OFFS1:MMM:VCO:OVER <NRf>`

`:SENS1:OFFS1:MMM:VCO:OVER?`

:SENSe{1-16}:OFFSet{1-50}:RCVSource:CW[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:RCVSource:CW[:STATe]?

Description: Sets the receiver source CW mode on/off status for the indicated multiple source band and channel. Outputs the receiver source CW mode on/off status for the indicated multiple source band and channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:RCVS:CW ON`

`:SENS1:OFFS1:RCVS:CW?`

:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:DIVisor?

Description: Sets the receiver source frequency divisor for the indicated multiple source band and channel. The multiple source equations depend on whether the instrument is in Broadband or Millimeter-Wave Mode.

- If the VNA is in Broadband Mode, the Start, Stop, and Broadband Break Frequencies in GHz set the frequency range.
- If the VNA is in Millimeter-Wave Mode, only the Start and Stop Frequency in GHz are required.

The resultant equations for LO and RF are:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Outputs the receiver source frequency divisor for the indicated multiple source band and channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: Depends on the multiple source equation.

Default Value: 1

Syntax Example:
```
:SENS1:OFFS1:RCVS:DIV 2
:SENS1:OFFS1:RCVS:DIV?
```

:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:MULTiplier <NRf>
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:MULTiplier?

Description: Sets the receiver source frequency multiplier for the indicated multiple source band and channel. Outputs the receiver source frequency multiplier for the indicated multiple source band and channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example:
```
:SENS1:OFFS1:RCVS:MULT 4
:SENS1:OFFS1:RCVS:MULT?
```

:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:OFFSet?

Description: Sets the receiver source offset frequency for the indicated multiple source band and channel. The multiple source equations depend on whether the instrument is in Broadband or Millimeter-Wave Mode.

- If the VNA is in Broadband Mode, the Start, Stop, and Broadband Break Frequencies in GHz set the frequency range.
- If the VNA is in Millimeter-Wave Mode, only the Start and Stop Frequency in GHz are required.

The resultant equations for LO and RF are:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Outputs the receiver source offset frequency for the indicated multiple source band and channel.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Depends on the multiple source equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS1:RCVS:OFFS 1.0E9`

`:SENS1:OFFS1:RCVS:OFFS?`

:SENSe{1-16}:OFFSet{1-50}:RECEiver:CW[:STATe] <char>
:SENSe{1-16}:OFFSet{1-50}:RECEiver:CW[:STATe]?

Description: Sets the receiver CW mode on/off status for the indicated multiple source band and channel. Outputs the receiver CW mode on/off status for the indicated multiple source band and channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:OFFS1:RECE:CW ON`

`:SENS1:OFFS1:RECE:CW?`

:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:DIVisor <NRf>
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:DIVisor?

Description: Sets the receiver frequency divisor for the indicated multiple source band and channel. Outputs the receiver frequency divisor for the indicated multiple source band and channel. The multiple source equations depend on whether the instrument is in Broadband or Millimeter-Wave Mode.

- If the VNA is in Broadband Mode, the Start, Stop, and Broadband Break Frequencies in GHz set the frequency range.
- If the VNA is in Millimeter-Wave Mode, only the Start and Stop Frequency in GHz are required.

The resultant equations for LO and RF are:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: Depends on the multiple source equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS1:RECE:DIV 2`

`:SENS1:OFFS1:RECE:DIV?`

:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:MULTiplier <NRf>
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:MULTiplier?

Description: Sets the receiver frequency multiplier for the indicated multiple source band and channel. Outputs the receiver frequency multiplier for the indicated multiple source band and channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:OFFS1:RECE:MULT 4`

`:SENS1:OFFS1:RECE:MULT?`

:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:OFFSet <NRf>
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:OFFSet?

Description: Sets the receiver offset frequency for the indicated multiple source band and channel. Outputs the receiver offset frequency for the indicated multiple source band and channel. The multiple source equations depend on whether the instrument is in Broadband or Millimeter-Wave Mode.

- If the VNA is in Broadband Mode, the Start, Stop, and Broadband Break Frequencies in GHz set the frequency range.
- If the VNA is in Millimeter-Wave Mode, only the Start and Stop Frequency in GHz are required.

The resultant equations for LO and RF are:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Depends on the multiple source equation.

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:OFFS1:RECE:OFFS 1.0E-4`

`:SENS1:OFFS1:RECE:OFFS?`

:SENSe{1-16}:OFFSet{1-50}:STARt <NRf>
:SENSe{1-16}:OFFSet{1-50}:STARt?

Description: Sets the start frequency of the indicated multiple source band and channel. Outputs the start frequency of the indicated multiple source band and channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: 7.00000000000E+004

Syntax Example: `:SENS1:OFFS1:STAR 2.0E9`

`:SENS1:OFFS1:STAR?`

:SENSe{1-16}:OFFSet{1-50}:STOP <NRf>
:SENSe{1-16}:OFFSet{1-50}:STOP?

Description: Sets the stop frequency of the indicated multiple source band and channel.

Outputs the stop frequency of the indicated multiple source band and channel.

The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options.

The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step Size) to Maximum Instrument Frequency

Default Value: 1.00000000000E+010

Syntax Example: `:SENS1:OFFS1:STOP 5.0E9`

`:SENS1:OFFS1:STOP?`

5-96 :SENSe{1-16}:RECEiver Subsystem

The :SENSe{1-16}:RECEiver subsystem commands are used to the configure the VNA receiver.

:SENSe{1-16}:RECeiver:CALibration:EXISt? <char>

Description: Checks for the existence of a particular receiver calibration on the indicated channel.

Cmd Parameters: NA

Query Parameters: <char> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:REC:CAL:EXIS? A1`

:SENSe{1-16}:RECeiver:CALibration:STATe <char>
:SENSe{1-16}:RECeiver:CALibration:STATe?

Description: Turns the receiver calibration on/off on the given channel. Query outputs the receiver calibration on/off status on the given channel.

Cmd Parameters: <char1> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4 | ALL

<char2> 1 | 0 | ON | OFF

Query Parameters: <char1> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:REC:CAL:STAT A1,ON`

`:SENS1:REC:CAL:STAT?A1`

:SENSe{1-16}:RECeiver:CALibration? <char>

Description: Performs the indicated receiver calibration on the indicated channel and outputs status.

Cmd Parameters: NA

Query Parameters: <char1> A1 | A2 | A3 | A4 | B1 | B2 | B3 | B4

<char2> PORT1 | PORT2 | PORT3 | PORT4

Output: <NR1 Data> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:REC:CAL? A1,PORT1`

:SENSe{1-16}:RECEiver:CONFiguration <char>
:SENSe{1-16}:RECEiver:CONFiguration?

Description: Sets the receiver configuration on the indicated channel. The command parameters available depend on the instrument series and model number, where:

- STANdard = The receiver frequency follows the source frequency.
- MSOURce = Multiple source receiver configuration. This can be used with or without actual multiple sources. If selected, allows the receiver frequency to be changed by an equation.
- MMWBBand = Millimeter-Wave broadband receiver configuration. Sets the receiver to be controlled by equations on the VNA appropriate for Broadband/ Millimeter-Wave systems.
- MODBBand = Modular Broadband receiver configuration. Only available on MS4647B VNAs equipped with Option 80/81.

MSOURce

For MSOURce (Multiple Source), the equation used depends on whether CW is OFF or ON.

- If CW is OFF, Source = (Multiplier / Divisor) × (Frequency + Offset Frequency)
- If CW is ON, Source = (Multiplier / Divisor) × Offset Frequency

MMWBBand

For MMWBBand (Millimeter-Wave Broadband), the equations used vary for the LO and RF frequencies:

- LO External Source 1 = (Multiplier / Divisor) × (F – 12.35 MHz)
- RF External Source 2 = (Multiplier / Divisor) × F

Outputs the receiver configuration on the indicated channel.

Cmd Parameters: The available command and query parameters vary depending on the instrument model and installed options:

- For MS4647B with Option 80/81 = <char> STANdard | MSOURce | MMWBBand | MODBBand
- For MS464xB without Option 80/81 = <char> STANdard | MSOURce | MMWBBand

Query Parameters: <char> STAN | MSOUR | MMWBB | MODBB

Range: NA

Default Value: NA

Syntax Example: `:SENS1:REC:CONF STAN`

`:SENS1:REC:CONF?`

5-97 :SENSe{1-16}:SEGMent Subsystem

The :SENSe{1-16}:SEGMent subsystem command is used to query the segmented sweep type as frequency-based or index based on the active sweep.

Limit Line and Segment Subsystems

Related limit line and segment configuration and control subsystems are:

- ":CALCulate{1-16}[:SELected]:LIMit Subsystem" on page 5-153
- ":DISPlay Subsystem" on page 5-223
- ":SENSe{1-16}:FSEGMent Subsystem" on page 5-401.
- ":SENSe{1-16}:FSEGMent{1-50} Subsystem" on page 5-412.
- ":SENSe{1-16}:ISEGMent Subsystem" on page 5-436.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446.
- ":SENSe{1-16}:ISEGMent{1-50} Subsystem" on page 5-446
- ":SENSe{1-16}:SEGMent Subsystem" on page 5-487

:SENSe{1-16}:SEGMent:TYPe?

Description: Query only. On the indicated channel, outputs the segmented sweep type for the active segment as frequency-based or index-based where:

- FREQ = Frequency-based sweep type
- INDEX = Index-based sweep type

Query Parameters: <char> FREQ | INDEX

Range: NA

Default Value: NA

Syntax Example: `:SENS1:SEGM:TYP?`

5-98 :SENSe{1-16}:SOURce{1-4} Subsystem

The :SENSe{1-16}:SOURce{1-4} subsystem commands are used to configure and control external sources.

Trigger, Hold, and External Source Subsystems

Related trigger, hold, and external source subsystems are:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419
- ":SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem" on page 5-452
- ":SENSe{1-16}:SOURce{1-4} Subsystem" on page 5-488
- ":SOURce:ALL:EXTernal Subsystem" on page 5-508
- ":SOURce{1-4}:EXTernal Subsystem" on page 5-526
- ":TRIGger[:SEQuence] Subsystem" on page 5-552

:SENSe{1-16}:SOURce{1-4}:EXTernal[:STATe] <char>
:SENSe{1-16}:SOURce{1-4}:EXTernal[:STATe]?

Description: Turns on/off the control of the indicated external source on the indicated channel. Requires from one to four external sources. Note that the command does not disable the RF output of the external source. Outputs the on/off status of control of the indicated external source on the indicated channel. Requires from one to four external sources.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:SOUR1:EXT ON`

`:SENS1:SOUR1:EXT?`

5-99 :SENSe{1-16}:SPUR Subsystem

The :SENSe{1-16}:SPUR subsystem command is used to toggle spur reduction on and off on the indicated channel.

Channel and Sweep Subsystems

Related channel and sweep configuration and control subsystems are:

- ":SENSe{1-16}:AVERage Subsystem" on page 5-265
- ":SENSe{1-16}:FREQuency Subsystem" on page 5-399
- ":SENSe{1-16}:SPUR Subsystem" on page 5-489
- ":SENSe{1-16}:SWEep Subsystem" on page 5-490

:SENSe{1-16}:SPUR:REDuction[:STATe] <char>
:SENSe{1-16}:SPUR:REDuction[:STATe]?

Description: Turns spur reduction on/off for the indicated channel. Outputs spur reduction on/off status for the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SENS1:SPUR:RED ON`

`:SENS1:SPUR:RED?`

5-100 :SENSe{1-16}:SWEep Subsystem

The :SENSe{1-16}:SWEep subsystem commands are used to configure and control the instrument sweeps on a channel by channel basis.

Channel and Sweep Subsystems

Related channel and sweep configuration and control subsystems are:

- ":SENSe{1-16}:AVERage Subsystem" on page 5-265
- ":SENSe{1-16}:FREQuency Subsystem" on page 5-399
- ":SENSe{1-16}:SPUR Subsystem" on page 5-489
- ":SENSe{1-16}:SWEep Subsystem" on page 5-490

:SENSe{1-16}:SWEep:CW:POINt <NRf>
:SENSe{1-16}:SWEep:CW:POINt?

Description: Sets the CW sweep mode number of points of the given channel. Outputs the CW sweep mode number of points of the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: The range depends on the maximum points setting:

- When 25,000 points is set, range = 2 to 25,000.
- When 100,000 points is set, range = 2 to 100,000.

Default Value: 201

Syntax Example: `:SENS1:SWE:CW:POIN 1.01E2`

`:SENS1:SWE:CW:POIN?`

:SENSe{1-16}:SWEep:CW[:STATe] <char>
:SENSe{1-16}:SWEep:CW[:STATe]?

Description: Turns on/off the CW sweep mode of the given channel. Outputs the on/off status of the CW sweep mode of the given channel

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: `:SENS1:SWE:CW 1`

`:SENS1:SWE:CW?`

:SENSe{1-16}:SWEep:DELay <NRf>
:SENSe{1-16}:SWEep:DELay?

Description: Sets the sweep delay time of the indicated channel. Outputs the sweep delay time of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 0 to 100

Default Value: 0.00000000000E+000

Syntax Example: `:SENS1:SWE:DEL 50.0E-3`

`:SENS1:SWE:DEL?`

:SENSe{1-16}:SWEep:DELay:TYPe <char>
:SENSe{1-16}:SWEep:DELay:TYPe?

Description: The command sets the sweep delay type of the indicated channel, as either Off, Load Pulse, or Phase Locked. The sweep delay type definitions are:

- LOAD = Load Pulse which sets the sweep so that the sweep starts, and while sweeping, the VNA digital signal processor sends a command to load the programming for the next frequency. At this point, the user-defined delay time period occurs. The sweep-load-delay cycle repeats for every point in the sweep.
- LOCK = Phase Locked which sets the sweep delay so that the sweep starts, the hardware frequency phase locks at the point, and then user-defined delay time period occurs. The sweep continues until the sweep is completed.
- OFF = The instrument sweeps normally without sweep delay.

The query outputs the sweep delay type of the indicated channel.

Cmd Parameters: <char> LOCK | LOAD | OFF

Query Parameters: <char> LOCK | LOAD | OFF

Range: NA

Default Value: OFF

Syntax Example: `:SENS1:SWE:DEL:TYP LOAD`

`:SENS1:SWE:DEL:TYP?`

:SENSe{1-16}:SWEep:POINt <NRf>
:SENSe{1-16}:SWEep:POINt?

Description: Sets the number of measurement points of the given channel. Outputs the number of measurement points of the given channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: Range depends on the maximum points setting:

- When 25,000 points is set, range = 2 to 25,000 points.
- When 100,000 points is set, range = 2 to 100,000 points.

Default Value: 201

Syntax Example: `:SENS1:SWE:POIN 1.01E2`

`:SENS1:SWE:POIN?`

:SENSe{1-16}:SWEep:TIMe <NRf>
:SENSe{1-16}:SWEep:TIMe?

Description: Sets the Sweep Time of the indicated channel. Outputs the Sweep Time of the indicated channel.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: MPND

Default: 0.00000000000E+000

Syntax Example: `:SENS1:SWE:TIM 5.0E1`

`:SENS1:SWE:TIM?`

:SENSe{1-16}:SWEep:TIMe:DISPlay[:STATe] <char>
:SENSe{1-16}:SWEep:TIMe:DISPlay[:STATe]?

Description: Sets the on/off state of the display of Sweep Time on the indicated channel. Outputs the on/off state the display of Sweep Time on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: `:SENS1:SWE:TIM:DISP 1`

`:SENS1:SWE:TIM:DISP?`

:SENSe{1-16}:SWEep:TIMe:STATe <char>
:SENSe{1-16}:SWEep:TIMe:STATe?

Description: Sets the on/off state of Sweep Time of the indicated channel. Outputs the on/off state of Sweep Time of the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default: 0

Syntax Example: `:SENS1:SWE:TIM:STAT 1`

`:SENS1:SWE:TIM:STAT?`

:SENSe{1-16}:SWEep:TIMe:TYPe <char>
:SENSe{1-16}:SWEep:TIMe:TYPe?

Description: Sets the Sweep Time Type of the indicated channel, where:

- AUTo = The VNA controls the sweep time automatically based on the sweep range and the number of sweep points.
- MANual = Allows the user to define the sweep time for the indicated channel.

Outputs the Sweep Time Type of the indicated channel.

Cmd Parameters: <char> AUTo | MANual

Query Parameters: <char> AUT | MAN

Range: NA

Default: AUT

Syntax Example: `:SENS1:SWE:TIM:TYP AUT`

`:SENS1:SWE:TIM:TYP?`

:SENSe{1-16}:SWEep:TYPe <char>
:SENSe{1-16}:SWEep:TYPe?

Description: Sets the sweep type of the indicated channel where the available sweep types are:

- LINear = Frequency-based linear sweep
- LOGarithmic = Frequency-based logarithmic sweep
- FSEGMent = Segment-based sweep with frequency-based segments
- ISEGMent = Index-based sweep with frequency-based segments
- POWer = Power-based sweep with either Power-based sweep with a CW frequency, or Power-based sweep with swept-frequency
- MFGC = Multiple frequency gain compression

Outputs the sweep type of the indicated channel.

Cmd Parameters: <char> LINear | LOGarithmic | FSEGMent | ISEGMent | POWer | MFGC

Query Parameters: <char> LIN | LOG | FSEGM | ISEGM | POW | MFGC

Range: NA

Default Value: LIN

Syntax Example: `:SENS1:SWE:TYP LIN`

`:SENS1:SWE:TYP?`

5-101 :SENSe{1-16}:TS3738 Subsystem

The :SENSe{1-16}:TS3738 subsystem commands are used to the configure and control the VectorStar ME7838A Broadband/Millimeter-Wave 3738A Test Set.

:SENSe{1-16}:TS3738:FREQuency:STARt <NRf>
:SENSe{1-16}:TS3738:FREQuency:STARt?

Description: The command sets the 3738A Test Set start frequency for the active channel. The query outputs the test set start frequency for the active channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: 6.70000000000E+010

Syntax Example: `:SENS1:TS3738:FREQ:STAR 5.0E9`

`:SENS1:TS3738:FREQ:STAR?`

:SENSe{1-16}:TS3738:FREQuency:STOP <NRf>
:SENSe{1-16}:TS3738:FREQuency:STOP?

Description: The command sets the 3738A Test Set stop frequency for the active channel. The query outputs the test set stop frequency for the active channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See "Minimum/Maximum Instrument Frequency and Related Parameters" on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step Size) to Maximum Instrument Frequency

Default Value: 1.10000000000E+011

Syntax Example: `:SENS1:TS3738:FREQ:STOP 1.0E10`

`:SENS1:TS3738:FREQ:STOP?`

:SENSe{1-16}:TS3738:LO:DIVisor <NRf>
:SENSe{1-16}:TS3738:LO:DIVisor?

Description: The command sets the test set LO divisor for the active channel. The query outputs the test set LO divisor for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 8

Syntax Example: `:SENS1:TS3738:LO:DIV 2`

`:SENS1:TS3738:LO:DIV?`

:SENSe{1-16}:TS3738:LO:MULTiplier <NRf>
:SENSe{1-16}:TS3738:LO:MULTiplier?

Description: The command sets the 3738 Test Set LO multiplier for the active channel. The query outputs the test set LO multiplier for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:TS3738:LO:MULT 2`

`:SENS1:TS3738:LO:MULT?`

:SENSe{1-16}:TS3738:OML:WGType <char>
:SENSe{1-16}:TS3738:OML:WGType?

Description: The command selects the 3738A Test Set OML waveguide type for the active channel. The available OML waveguide types are:

- WR01.5 = OML WM-380 waveguide rated for 500 to 750 GHz
- WR02.2 = OML WM-570 waveguide rated for 325 to 500 GHz
- WR03 = OML WM-864 waveguide rated for 220 to 325 GHz
- WR05 = OML WM-1295 waveguide rated for 140 to 220 GHz
- WR06 = OML WM-1651 waveguide rated for 110 to 170 GHz
- WR08 = OML WM-2032 waveguide rated for 90 to 140 GHz
- WR10 = OML WM-2540 waveguide rated for 75 to 110 GHz
- WR10E = OML WM-2540 waveguide rated for 67 to 110 GHz
- WR12 = OML WM-3175 waveguide rated for 60 to 90 GHz
- WR12E = OML WM-3175 waveguide rated for 56 to 94 GHz
- WR15 = OML WM-3810 waveguide rated for 50 to 75 GHz

The WR and WM designators are waveguide designators (not vendor specific) where WR is in english units (and is most common) and WM is in metric units. The WR number is approximately the waveguide broadwall dimension (in inches) times 10. The waveguide size uniquely sets the port frequency range of their model numbers.

The OML waveguide types are available from OML Inc., 300 Digital Drive, Morgan Hill, CA 95037, USA.

- Telephone: (408) 779-2698
- FAX: (408) 778-0491
- Web: http:// www.omlinc.com

The query outputs the test set OML waveguide type for the active channel.

Cmd Parameters: <char> WR01.5 | WR02.2 | WR03 | WR05 | WR06 | WR08 | WR10 | WR10E | WR12 | WR12E | WR15

Query Parameters: NA

Query Output: <char> WR01.5 | WR02.2 | WR03 | WR05 | WR06 | WR08 | WR10 | WR10E | WR12 | WR12E | WR15

Range: NA

Default Value: WR10E

Syntax Example: `:SENS1:TS3738:OML:WGT WR02.2`

`:SENS1:TS3738:OML:WGT?`

:SENSe{1-16}:TS3738:RECeiver <char>
:SENSe{1-16}:TS3738:RECeiver?

Description: The command selects the 3738A Test Set receiver type for the active channel where the available types are:

- MMWOML = Test set is using millimeter-wave modules from OML Inc.
- MMWVDI = Test set is using millimeter-wave modules from VDI Inc.
- BBAND = Test set is in broadband receiver type

The query outputs the 3738 Test Set receiver type for the active channel.

Cmd Parameters: <char> MMWOML | MMWVDI | BBAND

Query Parameters: NA

Query Output: <char> MMWOML | MMWVDI | BBAND

Range: NA

Default Value: NA

Syntax Example: `:SENS1:TS3738:REC MMWOML`

`:SENS1:TS3738:REC?`

:SENSe{1-16}:TS3738:RF:DIVisor <NRf>
:SENSe{1-16}:TS3738:RF:DIVisor?

Description: The command sets the 3738A Test Set RF divisor for the active channel. The query outputs the test set RF divisor for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 6

Syntax Example: `:SENS1:TS3738:RF:DIV 2`

`:SENS1:TS3738:RF:DIV?`

:SENSe{1-16}:TS3738:RF:MULTiplier <NRf>
:SENSe{1-16}:TS3738:RF:MULTiplier?

Description: The command sets the 3738A Test Set RF multiplier for the active channel. The query outputs the test set RF multiplier for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:TS3738:RF:MULT 2`

`:SENS1:TS3738:RF:MULT?`

:SENSe{1-16}:TS3738:VDI:MODel <char>
:SENSe{1-16}:TS3738:VDI:MODel?

Description: The command selects the 3738A Test Set VDI waveguide model number for the active channel. The parameters define different frequency plans for the same module band. The query outputs the test set VDI waveguide model for the active channel, where:

- MOD20G
- MOD24G

Cmd Parameters: <char> MOD20G | MOD24G

Query Parameters: NA

Query Output: <char> MOD20G | MOD24G

Range: NA

Default Value: MOD20G

Syntax Example: `:SENS1:TS3738:VDI:MOD MOD20G`

`:SENS1:TS3738:VDI:MOD?`

:SENSe{1-16}:TS3738:VDI:WGType <char>
:SENSe{1-16}:TS3738:VDI:WGType?

Description: The command selects the 3738A Test Set VDI waveguide type for the active channel. The available VDI waveguide types are:

- WR01.0 = VDI WM-250 for 750 to 1100 GHz
- WR01.5 = VDI WM-380 for 500 to 750 GHz
- WR02.2 = VDI WM-570 for 330 to 500 GHz
- WR03.4 = VDI WM-864 for 220 to 330 GHz
- WR04.3 = VDI WM-1092 for 170 to 260 GHz
- WR05.1 = VDI WM-1295 for 140 to 220 GHz
- WR06.5 = VDI WM-1651 for 110 to 170 GHz
- WR08 = VDI WM-2032 for 90 to 140 GHz
- WR10 = VDI WM-2540 for 75 to 110 GHz
- WR15 = VDI WM-3810 for 50 to 75 GHz

The WR and WM designators are waveguide designators (not vendor specific) where WR is in english units (and is most common) and WM is in metric units. The WR number is approximately the waveguide broadwall dimension (in inches) times 10. The waveguide size uniquely sets the port frequency range of their model numbers.

The VDI model numbers have suffix modifiers of -20G and -24G that are used to denote which frequency plan the VDI model is using. The suffix modifiers must be stated with the `:SENSe{1-16}:TS3738:VDI:MODel <char>` command described above.

The VDI waveguide types are from Virginia Diode, Inc. located at 979 Second Street, S.E., Suite 309, Charlottesville, VA 22902-6172, USA.

- Phone: 434.297.3257
- FAX: 434.297.3258
- Web: http://www.vadiodes.com

The query outputs the test set VDI waveguide type for the active channel.

Cmd Parameters: <char> WR01.0 | WR01.5 | WR02.2 | WR03.4 | WR04.3 | WR05.1 | WR06.5 | WR08 | WR10 | WR15

Query Parameters: NA

Query Output: <char> WR01.0 | WR01.5 | WR02.2 | WR03.4 | WR04.3 | WR05.1 | WR06.5 | WR08 | WR10 | WR15

Range: NA

Default Value: WR10

Syntax Example:
```
:SENS1:TS3738:VDI:WGT WR01.0
:SENS1:TS3738:VDI:WGT?
```

:SENSe{1-16}:TS3738[:STATe] <char>
:SENSe{1-16}:TS3738[:STATe]?

Description: The command sets the on/off status of the millimeter-wave or broadband mode on the indicated channel. The query outputs the on/off status of the millimeter-wave or broadband mode on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example:
```
:SENS1:TS3738 ON
:SENS1:TS3738?
```

5-102 :SENSe{1-16}:TS3739 Subsystem

The :SENSe{1-16}:TS3738 subsystem commands are used to the configure and control the VectorStar ME7838A Broadband/Millimeter-Wave 3738A Test Set.

:SENSe{1-16}:TS3739:FREQuency:STARt
:SENSe{1-16}:TS3739:FREQuency:STARt?

Description: The command sets the 3739x Test Set start frequency for the active channel. The query Outputs the test set start frequency for the active channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR3> The output parameter is in Hertz.

Range: Minimum Instrument Frequency to (Maximum Instrument Frequency minus Minimum Frequency Step Size)

Default Value: 6.70000000000E+010

Syntax Example: `:SENS1:TS3739:FREQ:STAR 5.0E9`

`:SENS1:TS3739:FREQ:STAR?`

:SENSe{1-16}:TS3739:FREQuency:STOP <NRf>
:SENSe{1-16}:TS3739:FREQuency:STOP?

Description: The command sets the 3739x Test Set stop frequency for the active channel. The query outputs the test set stop frequency for the active channel. The Minimum Instrument Frequency (Fmin) depends on the instrument installed options. The Maximum Instrument Frequency (Fmax) depends on the instrument model. See “Minimum/Maximum Instrument Frequency and Related Parameters” on page 1-22 for frequency limits for combinations of instrument model and available options. The Minimum Frequency Step Size is equal to 1 Hz.

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: (Minimum Instrument Frequency + Minimum Frequency Step Size) to Maximum Instrument Frequency

Default Value: 1.10000000000E+011

Syntax Example: `:SENS1:TS3739:FREQ:STOP 1.0E10`

`:SENS1:TS3739:FREQ:STOP?`

:SENSe{1-16}:TS3739:LO:DIVisor <NRf>
:SENSe{1-16}:TS3739:LO:DIVisor?

Description: The command sets the 3739x Test Set LO divisor for the active channel. The query Outputs the test set LO divisor for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is in Hertz.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 8

Syntax Example: `:SENS1:TS3739:LO:DIV 9`

`:SENS1:TS3739:LO:DIV?`

:SENSe{1-16}:TS3739:LO:MULTiplier <NRf>
:SENSe{1-16}:TS3739:LO:MULTiplier?

Description: The command sets the 3739x Test Set LO multiplier for the active channel. The query outputs the test set LO multiplier for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:TS3739:LO:MULT 2`

`:SENS1:TS3739:LO:MULT?`

:SENSe{1-16}:TS3739:OML:WGType
:SENSe{1-16}:TS3739:OML:WGType?

Description: The command selects the 3739x Test Set OML waveguide type for the active channel. The available OML waveguide types are:

- WR01.5 = OML WM-380 waveguide rated for 500 to 750 GHz
- WR02.2 = OML WM-570 waveguide rated for 325 to 500 GHz
- WR03 = OML WM-864 waveguide rated for 220 to 325 GHz
- WR05 = OML WM-1295 waveguide rated for 140 to 220 GHz
- WR06 = OML WM-1651 waveguide rated for 110 to 170 GHz
- WR08 = OML WM-2032 waveguide rated for 90 to 140 GHz
- WR10 = OML WM-2540 waveguide rated for 75 to 110 GHz
- WR10E = OML WM-2540 waveguide rated for 67 to 110 GHz
- WR12 = OML WM-3175 waveguide rated for 60 to 90 GHz
- WR12E = OML WM-3175 waveguide rated for 56 to 94 GHz
- WR15 = OML WM-3810 waveguide rated for 50 to 75 GHz

The WR and WM designators are waveguide designators (not vendor specific) where WR is in english units (and is most common) and WM is in metric units. The WR number is approximately the waveguide broadwall dimension (in inches) times 10. The waveguide size uniquely sets the port frequency range of their model numbers.

The OML waveguide types are available from OML Inc. http:// www.omlinc.com

The query outputs the test set OML waveguide type for the active channel.

Cmd Parameters: <char> WR01.5 | WR02.2 | WR03 | WR05 | WR06 | WR08 | WR10 | WR10E | WR12 | WR12E | WR15

Query Parameters: NA

Query Output: <char> WR01.5 | WR02.2 | WR03 | WR05 | WR06 | WR08 | WR10 | WR10E | WR12 | WR12E | WR15

Range: NA

Default Value: NA

Syntax Example:
```
:SENS1:TS3739:OML:WGT WR02.2
:SENS1:TS3739:OML:WGT?
```

:SENSe{1-16}:TS3739:RECeiver
:SENSe{1-16}:TS3739:RECeiver?

Description: The command selects the 3739x Test Set receiver type for the active channel where the available receiver types are:

- MMWOML = Test set for millimeter-wave modules from OML Inc.
- MMWVDI = Test set for millimeter-wave modules from VDI Inc.
- MODBB = Test set for modular broadband/millimeter-wave modules from Anritsu
- EBAND = Test set for millimeter-wave E-Band millimeter-wave module from Anritsu
- WBAND = Test set for W-Band millimeter-wave modules from Anritsu
- MODBB145 = Test set for modular broadband 145 GHz modules from Anritsu
- MODBANDED145 = Test set for modular banded 145 GHz modules from Anritsu

The query outputs the 3739x Test Set receiver type for the active channel.

Cmd Parameters: <char> MMWOML | MMWVDI | MODBB | EBAND | WBAND | MODBB145 | MODBANDED145

Query Parameters: NA

Query Output: <char> MMWOML | MMWVDI | MODBB | EBAND | WBAND | MODBB145 | MODBANDED145

Range: NA

Default Value: MODBB

Syntax Example: `:SENS1:TS3739:REC MODBB`

`:SENS1:TS3739:REC?`

:SENSe{1-16}:TS3739:RF:DIVisor <NRf>
:SENSe{1-16}:TS3739:RF:DIVisor?

Description: The command sets the test set RF divisor for the active channel. The query outputs the test set RF divisor for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 6

Syntax Example: `:SENS1:TS3739:RF:DIV 2`

`:SENS1:TS3739:RF:DIV?`

:SENSe{1-16}:TS3739:RF:MULTiplier <NRf>
:SENSe{1-16}:TS3739:RF:MULTiplier?

Description: Sets the test set RF multiplier for the active channel. Outputs the test set RF multiplier for the active channel. The command/query range is limited by the band equation which changes depending on the CW on/off mode:

- If CW is off: Source = (Multiplier/Divisor) × (Frequency + Offset Frequency)
- If CW is on: Source = (Multiplier/Divisor) + Offset Frequency

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: NA

Query Output: <NR1> The output parameter is an integer.

Range: MPNI. Limited by the band equation.

Default Value: 1

Syntax Example: `:SENS1:TS3739:RF:MULT 2`

`:SENS1:TS3739:RF:MULT?`

:SENSe{1-16}:TS3739:VDI:MODel
:SENSe{1-16}:TS3739:VDI:MODel?

Description: The command selects the test set VDI waveguide model for the active channel. The parameters define different frequency plans for the same module band. The query outputs the test set VDI waveguide model for the active channel, where:

- MOD20G
- MOD24G

The query outputs the test set VDI waveguide model for the active channel.

Cmd Parameters: <char> MOD20G | MOD24G

Query Parameters: NA

Query Output: <char> MOD20G | MOD24G

Range: NA

Default Value: MOD20G

Syntax Example: `:SENS1:TS3739:VDI:MOD MOD20G`

`:SENS1:TS3739:VDI:MOD?`

:SENSe{1-16}:TS3739:VDI:WGType <char>
:SENSe{1-16}:TS3739:VDI:WGType?

Description: The command selects the test set VDI waveguide type for the active channel. The available VDI waveguide types are:

- WR01.0 = VDI WM-250 for 750 to 1100 GHz
- WR01.5 = VDI WM-380 for 500 to 750 GHz
- WR02.2 = VDI WM-570 for 330 to 500 GHz
- WR03.4 = VDI WM-864 for 220 to 330 GHz
- WR04.3 = VDI WM-1092 for 170 to 260 GHz
- WR05.1 = VDI WM-1295 for 140 to 220 GHz
- WR06.5 = VDI WM-1651 for 110 to 170 GHz
- WR08 = VDI WM-2032 for 90 to 140 GHz
- WR10 = VDI WM-2540 for 75 to 110 GHz
- WR15 = VDI WM-3810 for 50 to 75 GHz

The WR and WM designators are waveguide designators (not vendor specific) where WR is in english units (and is most common) and WM is in metric units. The WR number is approximately the waveguide broadwall dimension (in inches) times 10. The waveguide size uniquely sets the port frequency range of their model numbers.

The VDI model numbers have suffix modifiers of -20G and -24G that are used to denote which frequency plan the VDI model is using. The suffix modifiers must be stated with the `:SENSe{1-16}:TS3739:VDI:MODel <char>` command described above.

The VDI waveguide types are from Virginia Diode, Inc. located at 979 Second Street, S.E., Suite 309, Charlottesville, VA 22902-6172, USA.

- Ph: 434.297.3257
- FAX: 434.297.3258
- Web: http://www.vadiodes.com

The query outputs the test set VDI waveguide type for the active channel. Cmd Parameters

<char> WR01.0 | WR01.5 | WR02.2 | WR03.4 | WR04.3 | WR05.1 | WR06.5 | WR08 | WR10 | WR15

Query Parameters: NA

Query Output: <char> WR01.0 | WR01.5 | WR02.2 | WR03.4 | WR04.3 | WR05.1 | WR06.5 | WR08 | WR10 | WR15

Range: NA

Default Value: WR10

Syntax Example: `:SENS1:TS3739:VDI:WGT WR05.1`

`:SENS1:TS3739:VDI:WGT?`

:SENSe{1-16}:TS3739[:STATe] <char>
:SENSe{1-16}:TS3739[:STATe]?

Description: The command sets the on/off status of the millimeter-wave or broadband mode on the indicated channel. The query outputs the on/off status of the millimeter-wave or broadband mode on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: :SENS1:TS3739

:SENS1:TS3739?

5-103 :SOURce:ALL:EXTernal Subsystem

The :SOURce:ALL:EXTernal subsystem command is used to control the state of fast triggering for external sources such as signal/function generators.

Trigger, Hold, and External Source Subsystems

Related trigger and hold subsystems are:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419
- ":SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem" on page 5-452
- ":SENSe{1-16}:SOURce{1-4} Subsystem" on page 5-488
- ":SOURce:ALL:EXTernal Subsystem" on page 5-508
- ":SOURce{1-4}:EXTernal Subsystem" on page 5-526
- ":TRIGger[:SEQuence] Subsystem" on page 5-552

:SOURce:ALL:EXTernal:FTRIGger[:STATe] <char>
:SOURce:ALL:EXTernal:FTRIGger[:STATe]?

Description: Turns on/off the fast trigger mode of the external sources. Outputs the on/off status of the fast trigger mode of the external sources.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR:ALL:EXT:FTRIG ON`

`:SOUR:ALL:EXT:FTRIG?`

5-104 :SOURce{1-16}:EFFective Subsystem

The :SOURce{1-16}:EFFective subsystem command is used to output the power level on the indicated channel and port.

Power Configuration Subsystems

Related power configuration and control systems are:

- ":SOURce{1-16}:EFFective Subsystem" on page 5-509
- ":SOURce{1-16}:POWer Subsystem" on page 5-511

:SOURce{1-16}:EFFective:MODBB:POWer:PORT{1-4}?

Description: Query only. The query outputs effective greater than 54 GHz power level of the given port on the given channel.

Cmd Parameters: NA

Query Parameters: NA

Output: <NR3> The output parameter is in dBm.

Range: –60 to +30 dBm

Default Value: The default value depends on model and installed options:

- MS4647B with Option 51, 61, or 62 = –10 dBm
- MS4647B without Option 51, 61, or 62 = –3 dBm

Syntax Example: `:SOUR1:EFF:MODBB:POW:PORT1?`

:SOURce{1-16}:EFFective:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]?

Description: Query only. Outputs the effective power level of the given port on the given channel.

Query Parameters: <NR3> The output parameter is in dBm.

Range: –30.0 to +30.0 dBm

Default Value: Default value depends on instrument model and installed options:

- For MS4645B or MS4647B with Option 51, 61, or 62 = –10 dBm
- For MS4645B or MS4647B without options = –3 dBm
- For the MS4642B or MS4644B = +5 dBm

Syntax Example: `:SOUR1:EFF:POW:PORT1?`

5-105 :SOURce{1-16}:MODBB Subsystem

The :SOURce{1-16}:EFFective subsystem command is used to output the power level on the indicated channel and port.

Power Configuration Subsystems

Related power configuration and control systems are:

- ":SOURce{1-16}:EFFective Subsystem" on page 5-509
- ":SOURce{1-16}:POWer Subsystem" on page 5-511

:SOURce{1-16}:MODBB:POWer:PORT{1-4} <NRf>
:SOURce{1-16}:MODBB:POWer:PORT{1-4}?

Description: The command sets the greater than 54 GHz power level of the given port on the given channel. The use of this command requires a MS4647B VNA equipped with Option 08x. The query outputs the greater than 54GHz power level of the given port on the given channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3> The output parameter is in dBm.

Range: –60 to +30 dBm

Default Value: The default value depends on model and installed options:

- MS4647B with Option 51, 61, or 62 = –10 dBm
- MS4647B without Option 51, 61, or 62 = –3 dBm

Syntax Example: `:SOUR1:MODBB:POW:PORT1 3.0E0`

`:SOUR1:MODBB:POW:PORT1?`

5-106 :SOURce{1-16}:POWer Subsystem

The :SOURce{1-16}:POWer subsystem commands are used configure and control port power types, attenuation types and levels, power coupling, calibration, and correction data for the indicated channel and port.

Power Configuration Subsystems

Related power configuration and control systems are:

- ":SOURce{1-16}:EFFective Subsystem" on page 5-509
- ":SOURce{1-16}:POWer Subsystem" on page 5-511 (this subsystem)

:SOURce{1-16}:POWer:COMBiner[:STATe] <char>
:SOURce{1-16}:POWer:COMBiner[:STATe]?

Description: The command enables or disables the RF combiner switch (Option 32) on the given channel.

The query outputs the state of the RF combiner switch on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:COMB:STAT ON`

`:SOUR1:POW:COMB:STAT?`

:SOURce{1-16}:POWer:PORT:COUPle <char>
:SOURce{1-16}:POWer:PORT:COUPle?

Description: Turns port power coupling on/off on the given channel.

Outputs the port power coupling on/off state on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 1

Syntax Example: `:SOUR1:POW:PORT:COUP ON`

`:SOUR1:POW:PORT:COUP?`

:SOURce{1-16}:POWer:PORT{1-7}:CORRection:COLLect

Description: Performs a flat power correction calibration for the indicated channel and port, and returns the calibration result. No query. The command returns status where:

- 0 = Calibration Passed
- 1 = Calibration Failed
- 2 = Calibration Aborted

For port definitions, see Table 5-1 on page 5-254.

Cmd Parameters: <NR1> 0 | 1 | 2

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:CORR:COLL`

:SOURce{1-16}:POWer:PORT{1-7}:CORRection:DATA <block>
:SOURce{1-16}:POWer:PORT{1-7}:CORRection:DATA?

Description: Inputs the flat power correction data for the indicated channel and port. Outputs the flat power correction data of the indicated channel and port.

For port definitions, see Table 5-1 on page 5-254.

Cmd Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Query Parameters: <block> data formatted as XML. See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:CORR:DATA <block>`

`:SOUR1:POW:PORT1:CORR:DATA?`

:SOURce{1-16}:POWer:PORT{1-7}:CORRection:TARGet <NRf>
:SOURce{1-16}:POWer:PORT{1-7}:CORRection:TARGet?

Description: Sets the power level target for the flat power correction calibration on the indicated channel and port. Outputs the power level target for the flat power correction calibration on the indicated channel and port.

For port definitions, see Table 5-1 on page 5-254.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: MPNF

Default Value: 0.000000E+000

Syntax Example: `:SOUR1:POW:PORT1:CORR:TARG 3.0`

`:SOUR1:POW:PORT1:CORR:TARG?`

:SOURce{1-16}:POWer:PORT{1-7}:CORRection[:STATe] <char>
:SOURce{1-16}:POWer:PORT{1-7}:CORRection[:STATe]?

Description: Turn on/off flat power correction for the indicated channel and port. Outputs the on/off status of flat power correction for the indicated channel and port.

For port definitions, see Table 5-1 on page 5-254.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:CORR ON`

`:SOUR1:POW:PORT1:CORR?`

:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:COLLect

Description: Performs a power linearity correction calibration for the indicated channel and port, and returns the calibration result. The command returns the calibration completion status where:

- 0 = Calibration Passed
- 1 = Calibration Failed
- 2 = Calibration Aborted

For port definitions, see Table 5-4.

Cmd Parameters: <NR1> 0 | 1 | 2

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:LIN:CORR:COLL`

Table 5-4. Port Number Definitions

Port Number	Definition
1	Port1 (2 and 4 port system)
2	Port2 (2 and 4 port system)
3	Src2 out Port1 (2 port system with option 32), (or Port3 on a 4 port system)
4	Port4 (4 port system)
5	Src2 out Port1 (4 port system)
6	Src2 out Port2 (4 port system)

:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:DATA <block>
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:DATA?

Description: Inputs the power sweep linearity calibration data for the indicated channel. Outputs the power sweep linearity calibration data of the indicated channel and port.

For port definitions, see Table 5-4 on page 5-513.

Cmd Parameters: <block> data formatted as XML. See definition of “<block> or <arbitrary block>” on page 2-12.

Query Parameters: <block> data formatted as XML. See definition of “<block> or <arbitrary block>” on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:LIN:CORR:DATA <block>`

`:SOUR1:POW:PORT1:LIN:CORR:DATA?`

:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:POWer:STARt <NRf>
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:POWer:STARt?

Description: Sets the power sweep start power target for the linear power calibration on the indicated channel and port. Outputs the power sweep start power target for the linear power calibration on the indicated channel and port.

For port definitions, see Table 5-4 on page 5-513.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: MPNF

Default Value: -2.000000E+001 dBm

Syntax Example: `:SOUR1:POW:PORT1:LIN:CORR:POW:STAR 2.3E1`

`:SOUR1:POW:PORT1:LIN:CORR:POW:STAR?`

:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:POWer:STOP?

Description: Query only. Outputs the power sweep start power target for the linear power calibration on the indicated channel and port.

For port definitions, see Table 5-4 on page 5-513.

Cmd Parameters: <NR3> The output parameter is in dBm.

Range: NA

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SOUR1:POW:PORT1:LIN:CORR:POW:STOP?`

:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection[:STATe] <char>
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection[:STATe]?

Description: The command turns on/off the power sweep linear calibration for the indicated channel and port. Outputs the on/off status of the power sweep linear calibrations for the indicated channel and port.

For port definitions, see Table 5-4 on page 5-513

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:LIN:CORR 1`

`:SOUR1:POW:PORT1:LIN:CORR?`

:SOURce{1-16}:POWer:PORT{1-4}:ATTenuation <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:ATTenuation?

Description: Sets the attenuation value of the indicated port on the indicated channel. Outputs the attenuation value of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: 0 to 60 dB in 10 dB increments

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:ATT 2E1`

`:SOUR1:POW:PORT1:ATT?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:EFFective:STOP?

Description: Query only. Outputs the power sweep effective (>54G) stop power of the indicated port on the indicated channel.

Cmd Parameters: NA

Query Parameters: NA

Output: <NR3>

Syntax Example: `:SOUR1:POW:PORT1:LIN:MODBB:POW:EFF:STOP?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STARt<NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STARt?

Description: The command sets the power sweep start power for frequencies greater than 54 GHz (>54G) on the indicated port on the indicated channel. The query outputs the power sweep (>54G) start power of the indicated port on the indicated channel.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Syntax Example: `:SOUR1:POW:PORT1:LIN:MODBB:POW:STAR <NRf>`

`:SOUR1:POW:PORT1:LIN:MODBB:POW:STAR?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STEP?

Description: Query only. The query outputs the power sweep (>54G) stepsize of the indicated port on the indicated channel for frequencies greater than 54 GHz.

Cmd Parameters: NA

Query Parameters: NA

Output: <NR3>

Syntax Example: `:SOUR1:POW:PORT1:LIN:MODBB:POW:STEP?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STOP <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STOP?

Description: The command sets the power sweep (>54G) stop power on the indicated port on the indicated channel for frequencies greater than 54 GHz. The query outputs the power sweep (>54G) stop power of the indicated port on the indicated channel for frequencies greater than 54 GHz.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Syntax Example: `:SOUR1:POW:PORT1:LIN:MODBB:POW:STOP <NRf>`

`:SOUR1:POW:PORT1:LIN:MODBB:POW:STOP?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:SINGle:POWer:EFFective:VALue?

Description: Query only. The query outputs the power sweep effective (>54G) single power value of the indicated port on the indicated channel for frequencies greater than 54 GHz.

Cmd Parameters:NA

Query Parameters:NA

Output:<NR3>

Syntax Example: `:SOUR1:POW:PORT1:LIN:MODBB:SING:POW:EFF:VAL?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:SINGle:POWer:VALue <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:SINGle:POWer:VALue?

Description: Sets the power sweep (>54G) single power value on the indicated port on the indicated channel for frequencies greater than 54 GHz. The query outputs the power sweep (>54G) single power value of the indicated port on the indicated channel for frequencies greater than 54 GHz.

Cmd Parameters: <NRf>

Query Parameters: NA

Output: <NR3>

Syntax Example: `:SOUR1:POW:PORT1:LIN:MODBB:SING:POW:VAL <NRf>`

`:SOUR1:POW:PORT1:LIN:MODBB:SING:POW:VAL?`

:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:EFFective:STARt?

Description: Query only. Outputs the power sweep effective start power of the indicated port on the indicated channel.

For port definitions, see Table 5-4 on page 5-513.

Query Parameters: <NR3> The output parameter is in dBm.

Range: MPNF

Default Value: The default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SOUR1:POW:PORT1:LIN:POW:EFF:STAR?`

:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:EFFective:STOP?

Description: Query only. Outputs the power sweep effective stop power of the indicated port on the indicated channel.

For port definitions, see Table 5-4 on page 5-513.

Query Parameters: <NR3> The output parameter is in dBm.

Range: MPNF

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SOUR1:POW:PORT1:LIN:POW:EFF:STOP?`

:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:OFFSet <NRf>
:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:OFFSet?

Description: Sets the power sweep offset power on the indicated port on the indicated channel. Outputs the power sweep offset power of the indicated port on the indicated channel.

For port definitions, see Table 5-4 on page 5-513.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: –1E2 to 1E2 dB

Default Value: 0.000000E+000

Syntax Example: `:SOUR1:POW:PORT1:LIN:POW:OFFS 1.2E1`

`:SOUR1:POW:PORT1:LIN:POW:OFFS?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:POINt <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:POINt?

Description: Sets the power sweep number of points on the indicated port on the indicated channel. Outputs the power sweep number of points of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 60

Default Value: 50

Syntax Example: `:SOUR1:POW:PORT1:LIN:POW:POIN 10E0`

`:SOUR1:POW:PORT1:LIN:POW:POIN?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STARt <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STARt?

Description: Sets the power sweep start power on the indicated port on the indicated channel. Outputs the power sweep start power of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: –3E1 to 2.99E1 dBm

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SOUR1:POW:PORT1:LIN:POW:STAR 2.10E1`

`:SOUR1:POW:PORT1:LIN:POW:STAR?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STEP?

Description: Query only. Outputs the power sweep step size of the indicated port on the indicated channel.

Query Parameters: <NR3> The output parameter is in dB.

Range: Power step is a calculated value based on the start and stop power levels divided by the number of points in the active channel.

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:LIN:POW:STEP?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STOP <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STOP?

Description: Sets the power sweep stop power on the indicated port on the indicated channel. Outputs the power sweep stop power of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: -2.99E1 to 3E1 dBm

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SOUR1:POW:PORT1:LIN:POW:STOP 2.5E1`

`:SOUR1:POW:PORT1:LIN:POW:STOP?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:REFerence:ATTenuation <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:REFerence:ATTenuation?

Description: Sets the power sweep reference attenuation value of the indicated port on the indicated channel. Outputs the power sweep reference attenuation value of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: 0 to 60 dB in 10 dB increments

Default Value: 0 dB

Syntax Example: `:SOUR1:POW:PORT1:LIN:REF:ATT 10E0`

`:SOUR1:POW:PORT1:LIN:REF:ATT?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer:EFFective:VALue?

Description: Query only. Outputs the power sweep effective single power value of the indicated port on the indicated channel.

Query Parameters: <NR3> The output parameter is in dBm.

Range: –20 dBm to +5 dBm

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:LIN:SING:POW:EFF:VAL?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer:VALue <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer:VALue?

Description: Sets the power sweep single power value on the indicated port on the indicated channel. Outputs the power sweep single power value of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: –3E1 to 3E1

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SOUR1:POW:PORT1:LIN:SING:POW:VAL 1.0E1`

`:SOUR1:POW:PORT1:LIN:SING:POW:VAL?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer[:STATe] <char>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer[:STATe]?

Description: Turns on/off the power sweep single power mode on the indicated port on the indicated channel. Outputs on/off state of the power sweep single power mode of the indicated port on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:LIN:SING:POW 1`

`:SOUR1:POW:PORT1:LIN:SING:POW?`

:SOURce{1-16}:POWer:PORT{1-4}:LINear:TEST:ATTenuation <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:LINear:TEST:ATTenuation?

Description: Sets the power sweep test attenuation value of the indicated port on the indicated channel. Outputs the power sweep test attenuation value of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: 0 to 60 dB in 10 dB increments

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:LIN:TEST:ATT 10E0`

`:SOUR1:POW:PORT1:LIN:TEST:ATT?`

:SOURce{1-16}:POWer:PORT{1-4}:REFerence:ATTenuation <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:REFerence:ATTenuation?

Description: Sets the reference attenuation value of the indicated port on the indicated channel. Outputs the reference attenuation value of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: 0 to 60 dB in 10 dB increments

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:ATT 2.0E1`

`:SOUR1:POW:PORT1:REF:ATT?`

:SOURce{1-16}:POWer:PORT{1-4}:SLOPe <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:SLOPe?

Description: Sets the power slope value of the indicated port on the indicated channel. Outputs the power slope value of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR3> The output parameter is a unitless number.

Range: –1E3 to 1E3

Default Value: 0.000000E+000

Syntax Example: `:SOUR1:POW:PORT1:SLOP 3.0E0`

`:SOUR1:POW:PORT1:SLOP?`

:SOURce{1-16}:POWer:PORT{1-4}:TEST:ATTenuation <NRf>
:SOURce{1-16}:POWer:PORT{1-4}:TEST:ATTenuation?

Description: Sets the test attenuation value of the indicated port on the indicated channel. Outputs the test attenuation value of the indicated port on the indicated channel.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR1> The output parameter is in dB.

Range: 0 to 60 dB in 10 dB increments

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:TEST:ATT 10E0`

`:SOUR1:POW:PORT1:TEST:ATT?`

:SOURce{1-16}:POWer:PORT{1-7}[:LEVel][:IMMediate][:AMPlitude] <NRf>
:SOURce{1-16}:POWer:PORT{1-7}[:LEVel][:IMMediate][:AMPlitude]?

Description: Sets the power level of the given port on the given channel. Outputs the power level of the given port on the given channel.

For port definitions, see Table 5-1 on page 5-254.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR3> The output parameter is in dB.

Range: –3E1 to 3E1

Default Value: Default value depends on model and installed options:

- MS4645B or MS4647B with Options 051, 061, or 062 = –10 dBm
- All other MS4645Bs and MS4647Bs = –3 dBm
- MS4642B and MS4644B = +5 dBm

Syntax Example: `:SOUR1:POW:PORT1 3.0E0`

`:SOUR1:POW:PORT1?`

:SOURce{1-16}:POWer:SLOPe[:STATe] <char>
:SOURce{1-16}:POWer:SLOPe[:STATe]?

Description: Turns on/off the power slope mode on the indicated channel. Outputs on/off state of the power slope mode on the indicated channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:SLOP ON`

`:SOUR1:POW:SLOP?`

5-107 :SOURce{1-16}:TS3739 Subsystem

The :SOURce{1-16}:TS3737 subsystem commands are used to configure and control the 3739x Broadband/Millimeter-Wave Test Set.indicated external source.

Related Subsystems

Related trigger, hold, and external source subsystems are:

- ":SENSe{1-16}:TS3739 Subsystem" on page 5-501

:SOURce{1-16}:TS3739:ALCCal:FREQuency:POWer:TABLe{1-10}:ENTRy <NRf>, <NRf>
:SOURce{1-16}:TS3739:ALCCal:FREQuency:POWer:TABLe{1-10}:ENTRy?

Description: The command sets a 3739 Test Set ALC calibration frequency/power table entry for the given channel. The query outputs a 3739 Test Set ALC calibration frequency/power table entry for the given channel. This calibration is performed over the operating frequency range of a Millimeter-Wave (mm-Wave) Test Set. See the types available from VDI or OML. The frequency values are constrained by the frequency range of the Test Set.

Best practices recommend that users know the power lever range of their Test Set and module as there will be no error thrown if the power level is set too high or too low. The power specifications of the mm-Wave modules should be well known, and it would be prudent if the user specified power levels in the range of the mm-Wave module he is using. One should also notice that the menu for this specifies that the power entered should be the maximum power for that module.

Note that when the VNA is in mm-Wave mode, pressing the front panel Preset (Reset) button takes VNA operation back to its regular mode with no mm-Wave operation and thus default values for this command do not exist.

Cmd Parameters: <NRf>, <NRf>

Where:

- <NRf> = Calibration frequency in Hertz.
- <NRf> = Power table entry in dBm.

Query Parameters: NA

Query Output: <NR3>, <NR3>

Where:

- <NR3 = Frequency in Hertz
- <NR3> = Power in dBm.

Range: Limited by VNA, test set, and mm-Wave module.

Default Value: NA

Syntax Example:

```
:SOUR1:TS3739:ALCC:FREQ:POW:TABL1:ENTR <NRf> <NRf>
:SOUR1:TS3739:ALCC:FREQ:POW:TABL1:ENTR?
```

:SOURce{1-16}:TS3739:ALCCal:LEVeltype <char>
:SOURce{1-16}:TS3739:ALCCal:LEVeltype?

Description: The command selects the 3739 Test Set ALC calibration leveling type for the given channel. The query outputs the 3739 Test Set ALC calibration leveling type for the given channel. The leveling types available are:

- IFLEVeling = For Modular BB, IF leveling provides the advantages of RF leveling described below plus the benefit of greater power control range and a lower minimum leveled power (at least 20 dB lower typically). As one approaches a 100 MHz difference between source and receiver frequencies, the power control range will start to decrease as will power control accuracy. The use of IF leveling is not recommended for source-receiver frequency differences greater than 100 MHz.
- RFLEVeling = For Modular BB, RF leveling provides a leveled RF output power with improved source match, protection against over powering the DUT, and insuring that the DUT is operating in its designated power range.
- VNALEVeling = Selecting VNALEVeling turns the RF or IF ALC off and the ALC accuracy is as it for the standalone VNA instrument.

Cmd Parameters: <char> IFLEVeling | RFLEVeling | VNALEVeling

Query Parameters: NA

Query Output: <char> IFLEV | RFLEV | VNALEV

Range: NA

Default Value: VNALEVeling

Syntax Example: `:SOUR1:TS3739:ALCC:LEV RFLEV`

`:SOUR1:TS3739:ALCC:LEV?`

:SOURce{1-16}:TS3739:ALCCal:POINt?

Description: Query only. Outputs the 3739 Test Set ALC calibration number of frequency/power points in the calibration. If setting up this calibration manually from the front panel, the number of points is set first and the user enters all the frequency/power values afterwards. The GPIB is the exact reverse. The user enters frequency, power pairs until they are satisfied or the maximum number of pairs reaches the limit of 25.

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1>

Range: 1 to 25 points

Default Value: 1

Syntax Example: `:SOUR1:TS3739:ALCC:POIN?`

:SOURce{1-16}:TS3739:ALCCal:PORT{1-4}:CORRection:COLLect?

Description: Query only. Performs a 3739 Test Set flat power correction calibration for the indicated channel and port and returns calibration status information, where:

- 0 = CalPass
- 1 = CalFail
- 2 = CalAbort
- 3 = CalNotExist
- 4 = CalGood

Cmd Parameters: NA

Query Parameters: NA

Query Output: <NR1>

Range: 0 to 4

Default Value: NA

Syntax Example: `:SOUR1:TS3739:ALCC:PORT1:CORR:COLL?`

5-108 :SOURce{1-4}:EXTernal Subsystem

The :SOURce{1-4}:EXTernal subsystem commands are used to configure and control the indicated external source.

Trigger, Hold, and External Source Subsystems

Related trigger, hold, and external source subsystems are:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419
- ":SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem" on page 5-452
- ":SENSe{1-16}:SOURce{1-4} Subsystem" on page 5-488
- ":SOURce:ALL:EXTernal Subsystem" on page 5-508
- ":SOURce{1-4}:EXTernal Subsystem" on page 5-526
- ":TRIGger[:SEQuence] Subsystem" on page 5-552

:SOURce{1-4}:EXTernal:ADDRess <NRf>
:SOURce{1-4}:EXTernal:ADDRess?

Description: Sets the GPIB address of the indicated external source. Outputs the GPIB address of the indicated external source.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 30

Default Value: The default value depends on which source is being used. The initial system default values are:

- External Source 1 = Address 4
- External Source 2 = Address 5
- External Source 3 = Address 2
- External Source 4 = Address 3

Syntax Example: `:SOUR1:EXT:ADDR 8`

`:SOUR1:EXT:ADDR?`

:SOURce{1-4}:EXTernal:FREQuency:MAXimum?

Description: Query only. Outputs the maximum frequency setting of the indicated external source.

Cmd Parameters: NA

Query Parameters: <NR3> The output parameter is in Hertz.

Range: NA

Default Value: Depends on the external source used.

Syntax Example: `:SOUR1:EXT:FREQ:MAX?`

:SOURce{1-4}:EXTernal:FREQuency:MINimum?

Description: Query only. Outputs the minimum frequency setting of the indicated external source.

Cmd Parameters: NA

Query Parameters: <NR3> The output parameter is in Hertz.

Range: NA

Default Value: Depends on the external source used.

Syntax Example: `:SOUR1:EXT:FREQ:MIN?`

:SOURce{1-4}:EXTernal:MODel?

Description: Query only. Outputs the model number of the indicated external source.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: Depends on the external source used.

Syntax Example: `:SOUR1:EXT:MOD?`

:SOURce{1-4}:EXTernal:POINt?

Description: Query only. Outputs the number of points setting of the indicated external source.

Query Parameters: <NR1> The output parameter is an integer.

Range: NA

Default Value: Depends on the external source used.

Syntax Example: `:SOUR1:EXT:POIN?`

:SOURce{1-4}:EXTernal:POWer <NRf>
:SOURce{1-4}:EXTernal:POWer?

Description: Sets the power level of the indicated external source. Outputs the power level of the indicated external source.

Cmd Parameters: <NRf> The input parameter is in dBm.

Query Parameters: <NR3> The output parameter is in dBm.

Range: NA

Default Value: Depends on the external source used.

Syntax Example: `:SOUR1:EXT:POW 3.0E0`

`:SOUR1:EXT:POW?`

5-109 :SOURce{1-16}:POWer Subsystem

The :SOURce{1-16}:POWer subsystem commands are used to configure and control the indicated source on the indicated port.

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed[:STATe] <char>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed[:STATe]?

Description: Turns on/off the power cal de-embedding operation using the S2P characterization data on the given channel. Outputs the on/off state of the power cal de-embedding operation using the S2P characterization data on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:NWE:FREQ:DEEM 1`

`:SOUR1:POW:PORT1:NWE:FREQ:DEEM?`

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed:FILe <string>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed:FILe?

Description: Sets the S2P characterization filename for performing power cal de-embedding operation on the given channel. Outputs the S2P characterization filename for performing power cal de-embedding operation on the given channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:NWE:FREQ:DEEM:FIL 'C:\filename.s2p'`

`:SOUR1:POW:PORT1:NWE:FREQ:DEEM:FIL?`

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed[:STATe] <char>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed[:STATe]?

Description: Turns on/off the power cal embedding operation using the S2P characterization data on the given channel. Outputs the on/off state of the power cal embedding operation using the S2P characterization data on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:NWE:FREQ:EMB 1`

`:SOUR1:POW:PORT1:NWE:FREQ:EMB?`

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed:FILe <string>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed:FILe?

Description: Sets the S2P characterization filename for performing power cal embedding operation on the given channel. Outputs the S2P characterization filename for performing power cal embedding operation on the given channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:NWE:FREQ:EMB:FIL 'C:\filename.s2p'`

`:SOUR1:POW:PORT1:NWE:FREQ:EMB:FIL?`

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed[:STATe] <char>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed[:STATe]?

Description: Turns on/off the power cal de-embedding operation using the S2P characterization data on the given channel. Outputs the on/off state of the power cal de-embedding operation using the S2P characterization data on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:NWE:POW:DEEM 1`

`:SOUR1:POW:PORT1:NWE:POW:DEEM?`

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed:FILe <string>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed:FILe?

Description: Sets the S2P characterization filename for performing power cal de-embedding operation on the given channel. Outputs the S2P characterization filename for performing power cal de-embedding operation on the given channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:NWE:POW:DEEM:FIL 'C:\filename.s2p'`

`:SOUR1:POW:PORT1:NWE:POW:DEEM:FIL?`

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed[:STATe] <char>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed[:STATe]?

Description: Turns on/off the power cal embedding operation using the S2P characterization data on the given channel. Outputs the on/off state of the power cal embedding operation using the S2P characterization data on the given channel.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: NA

Query Output: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:SOUR1:POW:PORT1:NWE:POW:EMB 1`

`:SOUR1:POW:PORT1:NWE:POW:EMB?`

:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed:FILe <string>
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed:FILe?

Description: Sets the S2P characterization filename for performing power cal embedding operation on the given channel. Outputs the S2P characterization filename for performing power cal embedding operation on the given channel.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.s2p' where x:\directory\ must exist. See Chapter 2, "Programming the VectorStar Series VNA", "Notational Conventions" on page 2-7 for more information.

Query Parameters: NA

Query Output: <char> Filename and path in the form: x:\directory\filename.s2p

Range: NA

Default Value: NA

Syntax Example: `:SOUR1:POW:PORT1:NWE:POW:EMB:FIL 'C:\filename.s2p'`

`:SOUR1:POW:PORT1:NWE:POW:EMB:FIL?`

5-110 :STATus:OPERation Subsystem

The :STATus:OPERation subsystem commands are used to set and output values from the Operation Status Enable Register.

:STATus:OPERation:CONDition?

Description: Query only. Outputs the value of the operation status condition register.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 32767

Default Value: 0

Syntax Example: `:STAT:OPER:COND?`

:STATus:OPERation:ENABle <NRf>
:STATus:OPERation:ENABle?

Description: Sets the value of the operation status enable register. Outputs the value of the operation status enable register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:OPER:ENAB 1`

`:STAT:OPER:ENAB?`

:STATus:OPERation:NTRansition <NRf>
:STATus:OPERation:NTRansition?

Description: Sets the value of the negative transition filter of the operation status register. Outputs the value of the negative transition filter of the operation status register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:OPER:NTR 1`

`:STAT:OPER:NTR?`

:STATus:OPERation:PTRansition <NRf>
:STATus:OPERation:PTRansition?

Description: Sets the value of the positive transition filter of the operation status register. Outputs the value of the positive transition filter of the operation status register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:OPER:PTR 1`

`:STAT:OPER:PTR?`

:STATus:OPERation[:EVENt]?

Description: Query only. Outputs the value of the operation status event register.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 32767

Default Value: 0

Syntax Example: `:STAT:OPER?`

5-111 :STATus:QUEStionable Subsystem

The :STATus:QUEStionable subsystem commands are used to set and output values from the Questionable Status Enable Register.

:STATus:QUEStionable:CONDition?

Description: Query only. Outputs the value of the questionable status condition register.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 32767

Default Value: 0

Syntax Example: `:STAT:QUES:COND?`

:STATus:QUEStionable:ENABle <NRf>
:STATus:QUEStionable:ENABle?

Description: Sets the value of the questionable status enable register. Outputs the value of the questionable status enable register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:QUES:ENAB 1`

`:STAT:QUES:ENAB?`

:STATus:QUEStionable:LIMit:CONDition?

Description: Query only. Outputs the value of the questionable limit status condition register.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 32767

Default Value: 0

Syntax Example: `:STAT:QUES:LIM:COND?`

:STATus:QUEStionable:LIMit:ENABle <NRf>
:STATus:QUEStionable:LIMit:ENABle?

Description: Sets the value of the questionable limit status enable register. Outputs the value of the questionable limit status enable register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:QUES:LIM:ENAB 1`

`:STAT:QUES:LIM:ENAB?`

:STATus:QUEStionable:LIMit:NTRansition <NRf>
:STATus:QUEStionable:LIMit:NTRansition?

Description: Sets the value of the negative transition filter of the questionable limit status register. Outputs the value of the negative transition filter of the questionable limit status register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:QUES:LIM:NTR 1`

`:STAT:QUES:LIM:NTR?`

:STATus:QUEStionable:LIMit:PTRansition <NRf>
:STATus:QUEStionable:LIMit:PTRansition?

Description: Sets the value of the positive transition filter of the questionable limit status register. Outputs the value of the positive transition filter of the questionable limit status register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: 0

Syntax Example: `:STAT:QUES:LIM:PTR 1`

`:STAT:QUES:LIM:PTR?`

:STATus:QUEStionable:LIMit[:EVENt]?

Description: Query only. Outputs the value of the questionable limit status event register.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 32767

Default Value: 0

Syntax Example: `:STAT:QUES:LIM?`

:STATus:QUEStionable:NTRansition <NRf>
:STATus:QUEStionable:NTRansition?

Description: Sets the value of the negative transition filter of the questionable status register. Outputs the value of the negative transition filter of the questionable status register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:QUES:NTR 1`

`:STAT:QUES:NTR?`

:STATus:QUEStionable:PTRansition <NRf>
:STATus:QUEStionable:PTRansition?

Description: Sets the value of the positive transition filter of the questionable status register. Outputs the value of the positive transition filter of the questionable status register.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 65535

Default Value: NA

Syntax Example: `:STAT:QUES:PTR 1`

`:STAT:QUES:PTR?`

:STATus:QUEStionable[:EVENt]?

Description: Query only. Outputs the value of the questionable status event register.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 32767

Default Value: 0

Syntax Example: `:STAT:QUES?`

5-112 :SYSTem Subsystem

The :SYSTEM subsystem commands configure and control various system-level instrument settings.

- :SYSTem:AUTOcal commands set the AutoCal Standard switch position.
- :SYSTem:COMMunicate commands are used to configure and control network communications for GPIB, TCP/IP, and USB networks.
- :SYSTem:DUT:PROTection command is used to toggle on and off the DUT protection state.
- :SYSTem:ERRor commands are used to query and clear the contents of the Error Queue.
- :SYSTem:EXTernal commands are used to query the status of bias tees, 4-Port test sets, and instrument model number.
- :SYSTem:HOLD command is used to RF hold status.
- :SYSTem:IFCalibration commands set the system IF calibration.
- :SYSTem:POFF command turns off the VNA.
- :SYSTem:POINt sets the maximum number of measurement points, and re-boots the instrument is required.
- :SYSTem:PORT returns the number of ports on the instrument.
- :SYSTem:PORT{1-4} commands set the attenuation for reference and test ports.
- :SYSTem:POWerup commands set the instrument power-up options and if required, set the power-up configuration file.
- :SYSTem:PRESet commands configure the instrument preset/reset configuration, and if required, set the preset/reset configuration file.
- :SYSTem:RETRace:RF command toggles the RF trace function on and off. The off state supports analysis of devices with AGC circuits.

:SYSTem:AUTocal:SWITch <char>
:SYSTem:AUTocal:SWITch?

Description: The command sets the AutoCal Standard switch position. The following switch positions are available where P1 = Test Port 1 and P2 = Test Port 2:

- SHSHort = P1 Short, P2 Short
- OPSHort = P1 Open, P2 Short
- SHOPen = P1 Short, P2 Open
- LD1SHort = P1 Load1, P2 Short
- SHLD1 = P1 Short, P2 Load1
- LD2SHort = P1 Load2, P2 Short
- SHLD2 = P1 Short, P2 Load 2
- LD3SHort = P1 Load 3, P2 Short
- SHLD3 = P1 Short, P2 Load 3
- THRu1 = P1 and P1 Through using Thru 1
- THRu2 = P1 and P2 Through using Thru 2
- THRu3 = P1 and P2 Through using Thru 3
- THRu4 = P1 and P2 Through using Thru 4
- OPENlight = Using Lightning 36581/36582 AutoCal, P1 Open and P2 Open
- SHORtlight = Using Lightning 36581/36582 AutoCal, P1 Short and P2 Short
- LOADlight = Using Lightning 36581/36582 AutoCal, P1 Load and P2 Short
- THRulight = Using Lighting 36581/36582 AutoCal, P1 and P2 Thru
- ASSURancelight = Using Lightning 36581/36582 AutoCal, Internal Assurance Standard

Outputs the AutoCal Standard switch position

Cmd Parameters: <char> SHSHort | OPSHort | SHOPen | LD1SHort | SHLD1 | LD2SHort | SHLD2 | LD3SHort | SHLD3 | THRu1 | THRu2 | THRu3 | THRu4 | OPENlight | SHORtlight | LOADlight | THRulight | ASSURancelight

Query Parameters: <char> SHSH | OPSH | SHOP | LD1SH | SHLD1 | LD2SH | SHLD2 | LD3SH | SHLD3 | THR1 | THR2 | THR3 | THR4 | OPEN | SHOR | LOAD | THR | ASSUR

Range: NA

Default Value: NA

Syntax Example: `:SYST:AUT:SWIT SHSHort`

`:SYST:AUT:SWIT?`

:SYSTem:COMMunicate:GPIB:ADDRess <NRf>
:SYSTem:COMMunicate:GPIB:ADDRess?

Description: Sets the GPIB address of the instrument’s GPIB remote interface. Outputs the GPIB address of the instrument’s GPIB remote interface.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 30

Default Value: 6

Syntax Example: `:SYST:COMM:GPIB:ADDR 7`

`:SYST:COMM:GPIB:ADDR?`

:SYSTem:COMMunicate:GPIB:BBANd:PMETer:ADDRess
:SYSTem:COMMunicate:GPIB:BBANd:PMETer:ADDRess?

Description: The command sets the GPIB address of the Broadband Power Sensor

The query outputs the GPIB address set for the Broadband Power Sensor

Cmd Parameters: <Integer> The input parameter is an integer.

Query Parameters: NA

Range: 1 to 30

Default Value: 20

Syntax Example: `:SYST:COMM:GPIB:PMET:BBAN:ADDR 20`

`:SYST:COMM:GPIB:PMET:BBAN:ADDR?`

:SYSTem:COMMunicate:GPIB:DBANd:PMETer:ADDRess <NRf>
:SYSTem:COMMunicate:GPIB:DBANd:PMETer:ADDRess?

Description: Sets the GPIB address of the attached D-Band Power Meter. Query outputs the GPIB address of the attached D-Band Power Meter.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 30

Default Value: 17

Syntax Example: `:SYST:COMM:GPIB:DBAN:PMET:ADDR 17`

`:SYST:COMM:GPIB:DBAN:PMET:ADDR?`

:SYSTem:COMMunicate:GPIB:EXTernal:TSET:ADDRess <NRf>
:SYSTem:COMMunicate:GPIB:EXTernal:TSET:ADDRess?

Description: Sets the GPIB address of the four port external test set. Outputs the GPIB address of the four port external test set.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 1 to 30

Default: 16

Syntax Example: `:SYST:COMM:GPIB:EXT:TSET:ADDR 16`

`:SYST:COMM:GPIB:EXT:TSET:ADDR?`

:SYSTem:COMMunicate:GPIB:FCOUNter:ADDRess <NRf>
:SYSTem:COMMunicate:GPIB:FCOUNter:ADDRess?

Description: Sets the GPIB address of the attached frequency counter. Outputs the GPIB address of the attached frequency counter.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 30

Default Value: 7

Syntax Example: `:SYST:COMM:GPIB:FCOUN:ADDR 9`

`:SYST:COMM:GPIB:FCOUN:ADDR?`

:SYSTem:COMMunicate:GPIB:PMETer:ADDRess <NRf>
:SYSTem:COMMunicate:GPIB:PMETer:ADDRess?

Description: Sets the GPIB address of the attached power meter. Outputs the GPIB address of the attached power meter.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 30

Default Value: 13

Syntax Example: `:SYST:COMM:GPIB:PMET:ADDR 6`

`:SYSTem:COMMunicate:GPIB:PMETer:ADDR?`

:SYSTem:COMMunicate:GPIB:PMETer:BBANd[:ADDRess]
:SYSTem:COMMunicate:GPIB:PMETer:BBANd[:ADDRess]?

Description: The command sets the GPIB address of the Broadband Power Sensor

The query outputs the GPIB address set for the Broadband Power Sensor

Cmd Parameters: <Integer> The input parameter is an integer.

Query Parameters: NA

Range: 1 to 30

Default Value: 20

Syntax Example: `:SYST:COMM:GPIB:PMET:BBAN 20`

`:SYST:COMM:GPIB:PMET:BBAN?`

:SYSTem:COMMunicate:GPIB:PMETer:DBANd[:ADDRess]
:SYSTem:COMMunicate:GPIB:PMETer:DBANd[:ADDRess]?

Description: The command sets the GPIB address of the D-Band Power Meter

The query outputs the GPIB address set for the D-Band Power Meter

Cmd Parameters: <Integer> The input parameter is an integer.

Query Parameters: NA

Range: 1 to 30

Default Value: 17

Syntax Example: `:SYST:COMM:GPIB:PMET:WBAN 17`

`:SYST:COMM:GPIB:PMET:WBAN?`

:SYSTem:COMMunicate:GPIB:PMETer:WBANd[:ADDRess]
:SYSTem:COMMunicate:GPIB:PMETer:WBANd[:ADDRess]?

Description: The command sets the GPIB address of the W-Band Power Meter

The query outputs the GPIB address set for the W-Band Power Meter

Cmd Parameters: <Integer> The input parameter is an integer.

Query Parameters: NA

Range: 1 to 30

Default Value: 15

Syntax Example: `:SYST:COMM:GPIB:PMET:WBAN 15`

`:SYST:COMM:GPIB:PMET:WBAN?`

:SYSTem:COMMunicate:GPIB:WBANd:PMETer:ADDRess <NRf>
:SYSTem:COMMunicate:GPIB:WBANd:PMETer:ADDRess?

Description: Sets the GPIB address of the attached W-Band Power Meter. Outputs the GPIB address of the attached W-Band Power Meter.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 30

Default Value: 15

Syntax Example: `:SYST:COMM:GPIB:WBAN:PMET:ADDR 6`

`:SYST:COMM:GPIB:WBAN:PMET:ADDR?`

:SYSTem:COMMunicate:TCPIP:ADDRess?

Description: Query only. Outputs the IP address of the Ethernet interface. The setting cannot be changed by the user.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: 0.0.0.0 to 255.255.255.255

Default Value: Varies with installation.

Syntax Example: `:SYST:COMM:TCPIP:ADDR?`

:SYSTem:COMMunicate:TCPIP:GATE?

Description: Query only. Outputs the default Gateway of the Ethernet interface. The setting cannot be changed by the user.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: 0.0.0.0 to 255.255.255.255

Default Value: Varies with installation.

Syntax Example: `:SYST:COMM:TCPIP:GATE?`

:SYSTem:COMMunicate:TCPIP:HDW?

Description: Query only. Outputs the MAC hardware address of the Ethernet interface. The setting cannot be changed by the user.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default Value: Varies with individual instrument.

Syntax Example: `:SYST:COMM:TCPIP:HDW?`

:SYSTem:COMMunicate:TCPIP:MASK?

Description: Query only. Outputs the instrument TCP/IP port address. The setting cannot be changed by the user.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0.0.0.0 to 255.255.255.255

Default Value: Varies with installation.

Syntax Example: `:SYST:COMM:TCPIP:MASK?`

:SYSTem:COMMunicate:TCPIP:PORT <NRf>
:SYSTem:COMMunicate:TCPIP:PORT?

Description: Enters the instrument TCP/IP port address. This value is user definable. The recommended TCP/IP address should be greater than or equal to 5001. Outputs the instrument TCP/IP port address.

Cmd Parameters: <NRf> The input parameter is a unitless number.

Query Parameters: <NR1> The output parameter is an integer.

Range: MPNI

Default Value: 5001

Syntax Example: `:SYST:COMM:TCPIP:PORT 5001`

`:SYST:COMM:TCPIP:PORT?`

:SYSTem:COMMunicate:USB:ADDRess?

Description: Query only. Outputs the address of the USB interface. The setting cannot be changed by the user.

Query Parameters: <char> The output parameter can be any combination of numbers and letters. The default values of "0B5B" and "FFD0" are fixed. The unit serial number, output as a Base 10 integer, varies with each instrument.

Range: NA

Default Value: NA

Syntax Example: `:SYST:COMM:USB:ADDR?`

:SYSTem:DUT:PROTection[:STATe] <char>
:SYSTem:DUT:PROTection[:STATe]?

Description: Sets the DUT Protection on/off state. Outputs the DUT Protection on/off state.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Syntax Example: `:SYST:DUT:PROT`

`:SYST:DUT:PROT?`

:SYSTem:ERRor:CLEar

Description: Clears the contents of the error queue. No query.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SYST:ERR:CLE`

:SYSTem:ERRor:COUNt?

Description: Query only. Outputs the number of errors in the error queue.

Query Parameters: <NR1> The output parameter is an integer.

Range: 0 to 100

Default Value: 0

Syntax Example: `:SYST:ERR:COUN?`

:SYSTem:ERRor:QUEue?

Description: Query only. Outputs the contents of the error queue.

Query Parameters: <block> See definition of "<block> or <arbitrary block>" on page 2-12.

Range: NA

Default Value: NA

Syntax Example: `:SYST:ERR:QUE?`

:SYSTem:ERRor[:NEXt]?

Description: Query only. Removes and outputs the oldest error in the error queue.

Query Parameters: <ASCII> See definition of "<ASCII> or <Arbitrary ASCII>" on page 2-12.

Range: NA

Default Value: No Error

Syntax Example: `:SYST:ERR?`

:SYSTem:EXTernal:TSET:FPORt:BIAS:TEES?

Description: Query only. The query outputs the existence of bias tees on the 4-Port Test Set where:

- No Bias Tees are installed = 0
- Bias Tees are installed = 1

Query Parameters: <char> 1 | 0

Range: NA

Default: NA

Syntax Example: `:SYST:EXT:TSET:FPOR:BIAS:TEES?`

:SYSTem:EXTernal:TSET:FPORt:EXISt?

Description: Query only. Outputs the 4-Port Test Set existence information where:

- MN4690B Series 4-Port Test Set does not exist = 0
- MN4690B Series 4-Port Test Set exists = 1

Query Parameters: <char> 1 | 0

Range: NA

Default: NA

Syntax Example: `:SYST:EXT:TSET:FPOR:EXIS?`

:SYSTem:EXTernal:TSET:FPORt:FREQuency:MAXimum?

Description: Query only. Outputs the MN4690B Series 4-Port Test Set maximum frequency value:

- MN4694B Series 4-Port Test Set = 40 GHz
- MN4697B Series 4-Port Test Set = 70 GHz

Query Parameters: <NR3> The output parameter is in Hertz.

Range: NA

Default: NA

Syntax Example: `:SYST:EXT:TSET:FPOR:FREQ:MAX?`

:SYSTem:EXTernal:TSET:FPORt:FREQuency:MINimum?

Description: Query only. Outputs the 4-Port Test Set minimum frequency value.

Query Parameters: <NR3> The output parameter is in Hertz.

Range: NA

Default: NA

Syntax Example: `:SYST:EXT:TSET:FPOR:FREQ:MIN?`

:SYSTem:EXTernal:TSET:FPORt:MODel?

Description: Query only. Outputs the model number of the 4-Port VNA instrument.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default: NA

Syntax Example: `:SYST:EXT:TSET:FPOR:MOD?`

:SYSTem:EXTernal:TSET:FPORt:REVision?

Description: Query only. Outputs the 4-Port Test Set revision number.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default: NA

Syntax Example: `:SYST:EXT:TSET:FPOR:REV?`

:SYSTem:EXTernal:TSET:FPORt:VERSion?

Description: Query only. Outputs the 4-Port Test Set version number.

Query Parameters: <char> The output parameter can be any combination of numbers and letters.

Range: NA

Default: NA

Syntax Example: `:SYST:EXT:TSET:FPOR:VERS?`

:SYSTem:HOLD:BIAS[:STATe] <char>
:SYSTem:HOLD:BIAS[:STATe]?

Description: Sets the Bias Tee on/off state in Hold. Outputs the Bias Tee on/off state in Hold

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SYST:HOLD:BIAS ON`

`:SYST:HOLD:BIAS?`

:SYSTem:HOLD:POWer <NRf>
:SYSTem:HOLD:POWer?

Description: The command sets the power sweep power level to use in Hold. The query outputs the power sweep power level to use in Hold.

Cmd Parameters: <NRf>: The parameter is in dBm.

Query Parameters: NA

Query Output: <NR3> The parameter is in dBm.

Range: NA

Default Value: NA

Syntax Example: `:SYST:HOLD:POW <NRf>`

`:SYST:HOLD:POW?`

:SYSTem:HOLD:RF[:STATe] <char>
:SYSTem:HOLD:RF[:STATe]?

Description: Sets the RF on/off state in Hold. Outputs the RF on/off state in Hold

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: NA

Syntax Example: `:SYST:HOLD:RF ON`

`:SYST:HOLD:RF?`

:SYSTem:IFCalibration:AUTomatic[:STATe] <char>
:SYSTem:IFCalibration:AUTomatic[:STATe]?

Description: Turns on/off the automatic occurrence of the IF calibration. Outputs the on/off state of the automatic occurrence of the IF calibration

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Syntax Example: `:SYST:IFC:AUT ON`

`:SYST:IFC:AUT?`

:SYSTem:IFCalibration:TRIGger

Description: Triggers an IF calibration. No query.

Cmd Parameters: NA

Syntax Example: `:SYST:IFC:TRIG`

:SYSTem:IFCalibration:DIGital:TRIGger

Description: Triggers a digital board IF calibration. Requires Option 35, IF Digitizer. No query.

Cmd Parameters: NA

Syntax Example: `:SYST:IFC:DIG:TRIG`

:SYSTem:POFF

Description: The command turns off the VNA. No query.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SYST:POFF`

:SYSTem:POINt:MAXimum <char>
:SYSTem:POINt:MAXimum?

Description: The command sets the maximum number of points the instrument can measure in a sweep. After changing the maximum number of points setting, an instrument reboot is required. After the command is issued and the maximum number of points changed, the instrument automatically reboots and then returns to its preset state.

VectorStar VNA Instruments in 100,000 Point Mode

If the instrument is set to 100,000 point mode:

- It will only support 1 channel and 16 traces.
- The buttons on the CHANNEL menu are not available.
- The CHANNEL drop-down menu on the Menu Bar is limited to non-channel commands.

VectorStar VNA Instruments in 25,000 Point Mode

If the instrument is in 25,000 point mode:

- It will support up to 16 channels, each with up to 16 traces.
- The CHANNEL menu buttons are available.
- The CHANNEL drop-down menu on the MENU BAR includes channel commands.

If the command requests the same number of points as the current instrument mode, no reboot is initiated.

*RST and Point Mode

The *RST command does not reset the instrument maximum point mode. For Syntax Example, when the instrument is in 100,000 point mode, the *RST command does not reset the number of points to 25,000.

Query Output

The query outputs current setting of the maximum number of points available in the instrument. If 25,000 is set, 25000 is returned. If 100,000 is set, 100000 is returned.

Cmd Parameters: <char> 25000 | 100000

Query Parameters: <char> 25000 | 100000

Range: NA

Default Value: 25000

Syntax Example: Instrument is running in factory default mode of up to 25,000 points with up to 16 channels, and up to 16 traces on each channel.

```
:SYST:POIN:MAX 100000
```

The instrument reboots itself. When the reboot is complete, the instrument is set to 100,000 maximum points with 1 channel and up to 16 traces.

```
:SYST:POIN:MAX?
```

```
100000
```

To return the instrument to 25,000 maximum points with up to 16 channels, each with up to 16 traces:

```
:SYST:POIN:MAX 25000
```

The instrument reboots itself. When the reboot is complete, the instrument is set to 25,000 maximum points, with up to 16 channels, each with up to 16 traces.

```
:SYST:POIN:MAX?
```

```
25000
```

If the command is issued for the same number of points as the current instrument mode, no reboot is provided.

```
:SYST:POIN:MAX?
```

```
25000
```

```
:SYST:POIN:MAX 25000
```

No reboot.

:SYSTem:PORT:COUNt?

Description: Query only. The query outputs the number of instrument test ports.

Cmd Parameters: NA

Query Parameters: <NR1> 1 | 2 | 3 | 4

Range: NA

Default Value: 2

Syntax Example: `:SYST:PORT:COUN?`

:SYSTem:PORT{1-4}:REFerence:ATTenuation <NRf>
:SYSTem:PORT{1-4}:REFerence:ATTenuation?

Description: The command sets the reference attenuation value on the given port. The query outputs the reference attenuation value on the given port. The VNA must have Option 61 or 62 installed.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR1> The output parameter is in dB.

Range: 0 to 60 dB in 10 dB increments

Default Value: 0 dB

Syntax Example: `:SYST:PORT1:REF:ATT 2.0E1`

`:SYST:PORT1:REF:ATT?`

:SYSTem:PORT{1-4}:TEST:ATTenuation <NRf>
:SYSTem:PORT{1-4}:TEST:ATTenuation?

Description: The command sets the test attenuation value on the given port. The query outputs the test attenuation value for the given port. The VNA must have Option 61 or 62 installed.

Cmd Parameters: <NRf> The input parameter is in dB.

Query Parameters: <NR1> The output parameter is in dB.

Range: 0 to 60 dB in 10 dB increments

Default Value: 0 dB

Syntax Example: `:SYST:PORT1:TEST:ATT 10E0`

`:SYST:PORT1:TEST:ATT?`

:SYSTem:POWerup:FILe <string>
:SYSTem:POWerup:FILe?

Description: The command sets the file path for the .cha power up configuration file to use on power up. The query outputs the file path of the .cha used for power up configuration.

Cmd Parameters: <string> Filename and path in the form: 'x:\directory\filename.cha' where x:\directory\filename.cha must exist. See definition of "<string>" on page 2-12.

Query Parameters: <char> Filename and path in the form: x:\directory\filename.cha

Range: NA

Default Value: NA

Syntax Example: `:SYST:POW:FIL 'C:\filepath\filename.cha'`

`:SYST:POW:FIL?`

:SYSTem:POWerup:TYPe <char>
:SYSTem:POWerup:TYPe?

Description: The command sets the power up instrument state to reset, last, or user defined. If USER is selected, a previously saved user-defined power up configuration file must exist and be stored on the instrument hard disk drive. The query outputs the power up instrument state as reset, last, or user defined.

Cmd Parameters: <char> RESET | LAST | USER

Query Parameters: <char> RESET | LAST | USER

Range: NA

Default Value: LAST

Syntax Example: `:SYST:POW:TYP RESET`

`:SYST:POW:TYP?`

:SYSTem:PRESet

Description: The command performs an instrument preset. This is the same function as provided by the front panel **Preset** key, or the ICON TOOLBAR | Preset icon, or the MENU BAR | Utilities | Preset command. No query. The type of preset performed depends on the type of preset set with the `:SYSTem:PREset:TYPE` command as either RESET or USER.

PRESet with RESET Set

If `RESET` has been is selected, the instrument returns to its factory as-shipped configuration, typically displaying one channel and four traces set to:

- Trace 1 (Tr1) set to S11 and a Smith Chart Impedance display.
- Trace 2 (Tr2) set to S12 and a dual rectangular display showing Log Magnitude and Phase graphs.
- Trace 3 (Tr3) set to S21 and a dual rectangular display showing Log Magnitude and Phase graphs.
- Trace 4 (Tr4) set to S22 and a Smith Chart Impedance display.
- Clears any user-defined segmented limit lines.

PRESET with USER Set

If `USER` has been selected, the instrument returns to a previously user-defined configuration state saved in either an All Channel Setup and Cal (CHA) file or in a All Channel Setup (STA) file.

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SYST:PRES`

:SYSTem:PRESet:FILe <string>
:SYSTem:PRESet:FILe?

Description: The commands sets the file path for the .cha file to use on preset. The query outputs the file path for the CHA file used for preset.

Cmd Parameters: <string> The directory and filename in the form: 'x:\directory\filename.cha' where x:\directory\filename.cha must exist. See definition of "<string>" on page 2-12.

Query Parameters: <char> Filename and path in the form: x:\directory\filename.cha

Range: NA

Default Value: NA

Syntax Example: `:SYST:PRES:FIL 'C:\directory\filename.cha'`

`:SYST:PRES:FIL?`

:SYSTem:PRESet:TYPe <char>
:SYSTem:PRESet:TYPe?

Description: The command sets the preset type to RESET, RESET0 (reset zero), or USER. The query outputs the selected preset type. The different preset types provide the following functions:

- USER - If the USER type is selected, a previously saved user-defined preset file must exist on the instrument disk drive. Once set, issuing a :SYSTem:PRESet command returns the instrument to the user-defined configuration. The user-defined preset file name is set using the :SYSTem:PRESet:FILe command.
- RESET - If the RESET type is selected, issuing a :SYSTem:PRESet command returns the instrument to the factory as-shipped configuration.
- RESET0 - If the RESET0 type is selected, issuing the :SYSTem:PRESet command clears the instrument memory of all calibration kit settings, calibrations, configurations, channel configurations, trace configurations, and other similar settings and returns the instrument to a factory as-shipped state. Functionally, issuing a :SYSTem:PRESet:TYPe RESET0 command followed by a :SYSTem:PRESet command is the same as issuing a :SYSTem:PRESet:ZERo command. Note that the RESET0 parameter does not delete any files. See the :SYSTem:PRESet:ZERo command description for a complete listing of included and excluded preset elements.

The preset type can also be configured by from the front panel by:

- Navigating to MAIN | System | SYSTEM | Setup | SETUP | Preset Setup | PRESET SETUP
- On the PRESET SETUP menu, selecting either Default, Default 0, or Saved Setup.
- If Saved Setup is to be used, the user-defined configuration file must be saved to the instrument solid-state drive. Then select the Select Saved Setup File button and select the desired user-defined preset file. Then select Saved Setup.

A preset can also be executed from the front panel by doing one of the following actions:

- Pressing the front panel **Preset key**.
- Selecting the ICON TOOLBAR | Preset icon.
- Selecting the MENU BAR | Utilities | Preset command.

Cmd Parameters: <char> RESET | RESET0 | USER

Query Parameters: <char> RESET | RESET0 | USER

Range: NA

Default Value: RESET

Syntax Example: :SYST:PRES:TYP RESET

:SYST:PRES:TYP?

:SYSTem:PRESet:ZERo

Description: The command performs a Reset Zero which does not delete any files, but clears the instrument memory of all calibration kit settings, calibrations, configurations, channel configurations, trace configurations, and other similar settings and returns the instrument to a factory as-shipped state described above in the command :SYSTem:PRESet on page 5-548. No query.

Similar to *RST

Similar to the *RST command, :SYST:PRES:ZER resets the instrument to a pre-defined condition with all user programmable command parameters set to their default values. These default parameter values are listed under each SCPI command in this manual.

Included Preset Conditions with :SYST:PRES:ZER

Except for the Command Parameters explicitly excluded in the next section, the :SYST:PRES:ZER command does the following:

- Sets the device-specific functions to a known state that is independent of the past-use history of the device;
- Device specific commands may be provided to program a different reset state than the original factory-supplied one;
- Clears any user-defined segmented limit lines;
- Sets the macro defined by *DDT to a device-defined state;
- Disables macros;
- Forces the device into the OCIS state (Operation Complete Command Idle State);
- Forces the device into the OQIS state (Operation Complete Query Idle State).

Excluded Preset Conditions with :SYST:PRES:ZER

The :SYST:PRES:ZER command DOES NOT make the following changes:

- Does not change the state of the IEEE 488.1 interface;
- Does not change the selected IEEE 488.1 address of the device;
- Does not change the Output Queue;
- Does not change any Event Enable Register settings including the Standard Event Status Enable Register settings;
- Does not change any Event Register setting including the Standard Event Status Resister settings;
- Does not change the power-on-status-clear flag setting; and
- Does not change the Service Request Enable Register.

Cmd Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:SYST:PRES:ZER`

:SYSTem:RETRace:RF[:STATe] <char>
:SYSTem:RETRace:RF[:STATe]?

Description: The command toggles the retrace RF energy state on or off. The default normal RF Retrace State is OFF.

RF Retrace OFF - Normal Operation

During normal operation, the VNA makes a sweep from the lowest to the highest point and the starts over again at the lowest point. During the start over or retrace period or when switching bands, the RF energy is blanked.

The best is to operate totally within an oscillator band so that the forward portion and retrace can step through the same frequency list either forward or backward and there is no loss of RF energy. With RF Retrace OFF, the VNA provides good leveling performance and spur rejection.

RF Retrace ON - AGC Circuits

For faster sweep measurement response and/or increased protection in DUTs with AGC (automatic gain control) circuits, this command sets the RF Retrace State to ON. With this setting, the RF energy status remains on during the sweep, during band changes, and during the sweep reset from highest point to lowest point.With RF Retrace ON, the VNA does not offer less leveling performance and spur rejection.

The query outputs the on/off state of retrace RF.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default: 1

Syntax Example: `:SYST:RETR:RF 0`

`:SYST:RETR:RF?`

5-113 :TRIGger[:SEQuence] Subsystem

The :TRIGger subsystem commands configure the trigger parameters and then control the trigger operation for internal, manual, and external trigger.

Trigger, Hold, and External Source Subsystems

Related trigger, hold, and external source subsystems are:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419
- ":SENSe{1-16}:OFFSet and :OFFset{1-50} Subsystem" on page 5-452
- ":SENSe{1-16}:SOURce{1-4} Subsystem" on page 5-488
- ":SOURce:ALL:EXTernal Subsystem" on page 5-508
- ":SOURce{1-4}:EXTernal Subsystem" on page 5-526
- ":TRIGger[:SEQuence] Subsystem" on page 5-552

:TRIGger[:SEQuence]:EXTernal:DELay <NRf>
:TRIGger[:SEQuence]:EXTernal:DELay?

Description: The command sets the delay time of the external trigger. The query outputs the delay time of the external trigger.

Cmd Parameters: <NRf> The input parameter is in Seconds.

Query Parameters: <NR3> The output parameter is in Seconds.

Range: 0 to 10 seconds with Minimum Step Size = 1 Nanosecond (ns). If delay is set below 1 ns, the instrument returns 0 (zero) seconds.

Default Value: 0.000000E+000

Syntax Example: `:TRIG:EXT:DEL 5.0E-2`

`:TRIG:EXT:DEL?`

:TRIGger[:SEQuence]:EXTernal:EDGe <char>
:TRIGger[:SEQuence]:EXTernal:EDGe?

Description: The command sets the leading/trailing edge triggering of the external trigger. The query outputs the leading/trailing edge triggering of the external trigger.

Cmd Parameters: <char> POSitive | NEGative

Query Parameters: <char> POS | NEG

Range: NA

Default Value: POS

Syntax Example: `:TRIG:EXT:EDG POS`

`:TRIG:EXT:EDG?`

:TRIGger[:SEQuence]:EXTernal:HANDshake[:STATe] <char>
:TRIGger[:SEQuence]:EXTernal:HANDshake[:STATe]?

Description: The command sets the on/off status of the external trigger handshake. The query outputs the on/off status of the external trigger handshake.

Cmd Parameters: <char> 1 | 0 | ON | OFF

Query Parameters: <char> 1 | 0

Range: NA

Default Value: 0

Syntax Example: `:TRIG:EXT:HAND ON`

`:TRIG:EXT:HAND?`

:TRIGger[:SEQuence]:EXTernal:TYPe <char>
:TRIGger[:SEQuence]:EXTernal:TYPe?

Description: The command sets the type of trigger that will be associated with the external trigger. The query outputs the type of trigger that will be associated with the external trigger.

Cmd Parameters: <char> POINt | SWEep | CHANnel | ALL

Query Parameters: <char> POIN | SWE | CHAN | ALL

Range: NA

Default Value: CHAN

Syntax Example: `:TRIG:EXT:TYP CHAN`

`:TRIG:EXT:TYP?`

:TRIGger[:SEQuence]:MANual:TYPe <char>
:TRIGger[:SEQuence]:MANual:TYPe?

Description: The command sets the type of trigger that will be associated with the manual trigger. The query outputs the type of trigger that will be associated with the manual trigger.

Cmd Parameters: <char> POINt | SWEep | CHANnel | ALL

Query Parameters: <char> POIN | SWE | CHAN | ALL

Range: NA

Default Value: CHAN

Syntax Example: `:TRIG:MAN:TYP SWE`

`:TRIG:MAN:TYP?`

:TRIGger[:SEQuence]:OUT[:STATe] <char>
:TRIGger[:SEQuence]:OUT[:STATe]?

Description: Turns on/off the trigger out. Query outputs the state of the trigger out.

Cmd Parameters: <char> 1|0|ON|OFF

Query Parameters: <char> 1|0

Range: NA

Default Value: 0

Syntax Example: `:TRIG:OUT ON`

`:TRIG:OUT?`

:TRIGger[:SEQuence]:REMote:TYPe <char>
:TRIGger[:SEQuence]:REMote:TYPe?

Description: The command sets the type of trigger that will be associated with the remote trigger. The query outputs the type of trigger that will be associated with the remote trigger.

Cmd Parameters: <char> POINt | SWEep | CHANnel | ALL

Query Parameters: <char> POIN | SWE | CHAN | ALL

Range: NA

Default Value: CHAN

Syntax Example: `:TRIG:REM:TYP POIN`

`:TRIG:REM:TYP?`

:TRIGger[:SEQuence]:SEDTransfer[:STATe] <char>
:TRIGger[:SEQuence]:SEDTransfer[:STATe]?

Description: The command Turns on/off the state of the data transfer at sweep end.

The query outputs the state of the data transfer at sweep end.

Cmd Parameters: <char> ON|OFF|1|0

Query Parameters: NA

Range: NA

Default Value: 0

Syntax Example: `:TRIG:SEDT ON`

`:TRIG:SEDT?`

:TRIGger[:SEQuence]:SOURce <char>
:TRIGger[:SEQuence]:SOURce?

Description: The command sets the source of sweep/measurement triggering where the following options are available:

- AUTO = The triggering source is automatic.
- MANual = The triggering source is manual.
- EXTTogpib = The triggering source is from the VNA Rear Panel BNC connector to the GPIB parser which handles it as a GPIB Group Execute Trigger. This triggering mode is typically used when an external device is used to trigger the VNA directly instead of over the GPIB bus.
- EXTernal = The triggering source is from the rear panel external trigger port.
- REMote = The triggering source is provided remotely.

The query outputs the source of the sweep/measurement triggering.

Cmd Parameters: <char> AUTO | MANual | EXTTogpib | EXTernal | REMote

Query Parameters: <char> AUTO | MAN | EXTT | EXT | REM

Range: NA

Default Value: AUTO

Syntax Example: `:TRIG:SOUR AUTO`

`:TRIG:SOUR?`

:TRIGger[:SEQuence][:IMMediate][:REMote]

Description: The command triggers a continuous sweep from the remote interface. During the sweep, command execution continues and does not pause. The operation of this command is modified by the instrument state set by the :SENse:HOLD:FUNCtion and/or the :SENSE{1-16}:HOLD:FUNCtion commands. See the following subsystems for additional information:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419

No query.

The operation of this command depends on the settings of the :SENSe:HOLD:FUNCtion, :TRIGger[:SEQuence][:REMote]:SINGle, and :TRIGger[:SEQuence][:IMMediate][:REMote] commands. Each setting combination is described in the sections below.

:SENSe:HOLD:FUNC CONT and :TRIG

```
:SENSe:HOLD:FUNCtion CONTinuous
// Sweep State = The sweep is sweeping continuously
// Command Execution = The parser is ready for a command right
away.
:TRIGger[:SEQuence][:IMMediate][:REMote]
// Sweep State = The sweep restarts and sweeps continuously. When
the sweep gets to the end of the sweep, it continues to sweep.
There is NO STATUS information that the end of the sweep has been
reached.
// Command Execution = The parser is ready for a command right away
```

:SENSe:HOLD:FUNC CONT and :TRIG:SING

```
:SENSe:HOLD:FUNCtion CONTinuous
// Sweep State = The sweep is sweeping continuously.
// Command Execution = The parser is ready for a command right away
:TRIGger[:SEQuence][:REMote]:SINGle
// Sweep State = The sweep restarts and sweeps continuously. When
the sweep gets to the end of the sweep, it sets the end of sweep
status bit and continues to sweep.
// Command Execution = Further execution is blocked until the end
of the sweep.
// Command Execution resumes when the sweep has reached the end of
the sweep.
```

:SENSe:HOLD:FUNC HOLD and :TRIG

```
:SENSe:HOLD:FUNCtion HOLD
// Sweep State = The sweep is stopped.
// Command Execution = The parser is ready for a command right away
:TRIGger[:SEQuence][:IMMediate][:REMote]
// Sweep State = The command has no effect. The sweep is stopped.
// Command Execution = The parser is ready for a command right away
```

:SENSe:HOLD:FUNC HOLD and :TRIG:SING

```
:SENSe:HOLD:FUNCtion HOLD
// Sweep State = The sweep is stopped
// Command Execution = The parser is ready for a command right away.
:TRIGger[:SEQuence][:REMote]:SINGle
// Sweep State = The sweep restarts and sweeps until the end of the sweep, at which point it sets the end of sweep status bit and stops.
// Command Execution = Further execution is blocked until the end of the sweep.
// Command Execution resumes when the sweep has reached the end of the sweep.
```

:SENSe:HOLD:FUNC SING and :TRIG

```
:SENSe:HOLD:FUNCtion SINGle
// Sweep State = The sweep does one complete sweep, goes into hold and stops.
// Command Execution = The parser is ready for a command.
:TRIGger[:SEQuence][:IMMediate][:REMote]
// Sweep State = The command has no effect. The sweep is stopped.
// Command Execution = The parser is ready for a command.
```

:SENSe:HOLD:FUNC SING and :TRIG:SING

```
:SENSe:HOLD:FUNCtion SINGle
// Sweep State = The sweep does one complete sweep, goes into hold and stops.
// Command Execution = The parser is ready for a command right away.
:TRIGger[:SEQuence][:REMote]:SINGle
// Sweep State = The sweep restarts and sweeps until the end of the sweep, at which point it sets the end of sweep status bit and stops.
// Command Execution = Further execution is blocked until the end of the sweep.
// Command Execution resumes when the sweep has reached the end of the sweep.
```

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:TRIG`

:TRIGger[:SEQuence][:REMote]:SINGle

Description: The command triggers a single sweep with synchronization from the remote interface. No query. During the sweep, command execution pauses until the sweep is complete. The operation of this command is modified by the instrument state set by the :SENse:HOLD:FUNCtion and/or the :SENSE{1-16}:HOLD:FUNCtion commands. See the following subsystems for additional information:

- ":SENSe:HOLD Subsystem" on page 5-261
- ":SENSe{1-16}:HOLD Subsystem" on page 5-419

Variable Operation

The operation of this command depends on the settings of the :SENSe:HOLD:FUNCtion, :TRIGger[:SEQuence][:REMote]:SINGle, and :TRIGger[:SEQuence][:IMMediate][:REMote] commands. Each setting combination is described in the sections below.

:SENSe:HOLD:FUNC CONT and :TRIG

```
:SENSe:HOLD:FUNCtion CONTinuous
// Sweep State = The sweep is sweeping continuously
// Command Execution = The parser is ready for a command right away.
:TRIGger[:SEQuence][:IMMediate][:REMote]
// Sweep State = The sweep restarts and sweeps continuously. When the sweep gets to the end of the sweep, it continues to sweep. There is NO STATUS information that the end of the sweep has been reached.
// Command Execution = The parser is ready for a command right away
```

:SENSe:HOLD:FUNC CONT and :TRIG:SING

```
:SENSe:HOLD:FUNCtion CONTinuous
// Sweep State = The sweep is sweeping continuously.
// Command Execution = The parser is ready for a command right away
:TRIGger[:SEQuence][:REMote]:SINGle
// Sweep State = The sweep restarts and sweeps continuously. When the sweep gets to the end of the sweep, it sets the end of sweep status bit and continues to sweep.
// Command Execution = Further execution is blocked until the end of the sweep.
// Command Execution resumes when the sweep has reached the end of the sweep.
```

:SENSe:HOLD:FUNC HOLD and :TRIG

```
:SENSe:HOLD:FUNCtion HOLD
// Sweep State = The sweep is stopped.
// Command Execution = The parser is ready for a command right away
:TRIGger[:SEQuence][:IMMediate][:REMote]
// Sweep State = The command has no effect. The sweep is stopped.
// Command Execution = The parser is ready for a command right away
```

:SENSe:HOLD:FUNC HOLD and :TRIG:SING

```
:SENSe:HOLD:FUNCtion HOLD

// Sweep State = The sweep is stopped

// Command Execution = The parser is ready for a command right
away.

:TRIGger[:SEQuence][:REMote]:SINGle

// Sweep State = The sweep restarts and sweeps until the end of the
sweep, at which point it sets the end of sweep status bit and
stops.

// Command Execution = Further execution is blocked until the end
of the sweep.

// Command Execution resumes when the sweep has reached the end of
the sweep.
```

:SENSe:HOLD:FUNC SING and :TRIG

```
:SENSe:HOLD:FUNCtion SINGle

// Sweep State = The sweep does one complete sweep, goes into hold
and stops.

// Command Execution = The parser is ready for a command.

:TRIGger[:SEQuence][:IMMediate][:REMote]

// Sweep State = The command has no effect. The sweep is stopped.

// Command Execution = The parser is ready for a command.
```

:SENSe:HOLD:FUNC SING and :TRIG:SING

```
:SENSe:HOLD:FUNCtion SINGle

// Sweep State = The sweep does one complete sweep, goes into hold
and stops.

// Command Execution = The parser is ready for a command right
away.

:TRIGger[:SEQuence][:REMote]:SINGle

// Sweep State = The sweep restarts and sweeps until the end of the
sweep, at which point it sets the end of sweep status bit and
stops.

// Command Execution = Further execution is blocked until the end
of the sweep.

// Command Execution resumes when the sweep has reached the end of
the sweep.
```

Cmd Parameters: NA

Query Parameters: NA

Range: NA

Default Value: NA

Syntax Example: `:TRIG:SING`

Appendix A — Programming with LabVIEW

A-1 Introduction

This document provides an overview of programming techniques for controlling the VectorStar VNA using the MS464xB LabVIEW driver over GPIB, TCP/IP (using VXI-11), and USB. This document assumes some previous knowledge of LabVIEW.

A-2 Overview

Programming Basics

There are many cases where a user may want programmatic control of the VNA. Examples include automating a test sequence, manufacturing testing, orchestrating a complex measurement involving various pieces of test equipment, gathering a time series of data, or as a convenient way of getting data, files or images transferred from the VNA to a pc for further analysis.

GPIB Mnemonics

VectorStar has a fairly large set of GPIB commands. This includes a set of Native commands, Lightning commands and Agilent 8510 commands. You use the same commands regardless of the communication method employed. See Example 4 for using the LIST command to output the full set of supported GPIB mnemonics directly from the VNA. For LabVIEW programmers the best way to contol VectorStar is with the native MS464X LabVIEW driver. But if you're familiar with Lightning, VectorStar also supports Lightning commands you can use the Lightning 37XXX LabVIEW driver to control VectorStar. This document uses a combination of MS464X driver VIs and VISA Reads and Writes to send commands to the VNA and to get data from the VNA. With this combination method, we use the driver to accomplish many procedures.

What is VISA?

VISA (Virtual Instrument System Architecture) is an I/O software standard for communicating with test instruments like VectorStar over any of the bus architectures which VectorStar supports. A VISA driver is available from both National Instruments and Agilent. National Instruments VISA drivers are available for the following operating systems: Windows, Mac OS X, Linux and others. It's always a good idea to get the latest driver (Version 5.1.1), but make sure to get the Full Version (not just the runtime) for the best support of the latest .NET Framework, USB and TCP/IP. The driver is available from http://www.ni.com/visa/ or from NI Device Driver CD that comes with NI hardware and is installed along with LabVIEW. The MS464X LabVIEW drivers uses VISA to communicate.

VISA uses connection strings to set up communication with the VNA over various protocols. Here are some connection string examples:

```
//VXI-11 Connection string
"TCPIP0::192.168.1.7::INSTR";

//GPIB Connection string
"GPIB0::6::INSTR";

//USB Connection string (vendor::product::serial_number)
"USB0::0x0B5B::0xFFD0::MS4647B-12345::INSTR";
```

The beauty of using VISA is that the only thing that needs to be changed for any of these possible communication protocols is the connection string. The rest of the code should be exactly the same (except for SOCKETS which are not covered in this document). For TCP/IP we recommend using VXI-11 since it better implements the IEEE 488.2 standard and all status checking. The *Getting Started* section of the Programming Examples shows how to set up for communication over VXI-11 (TCP/IP).

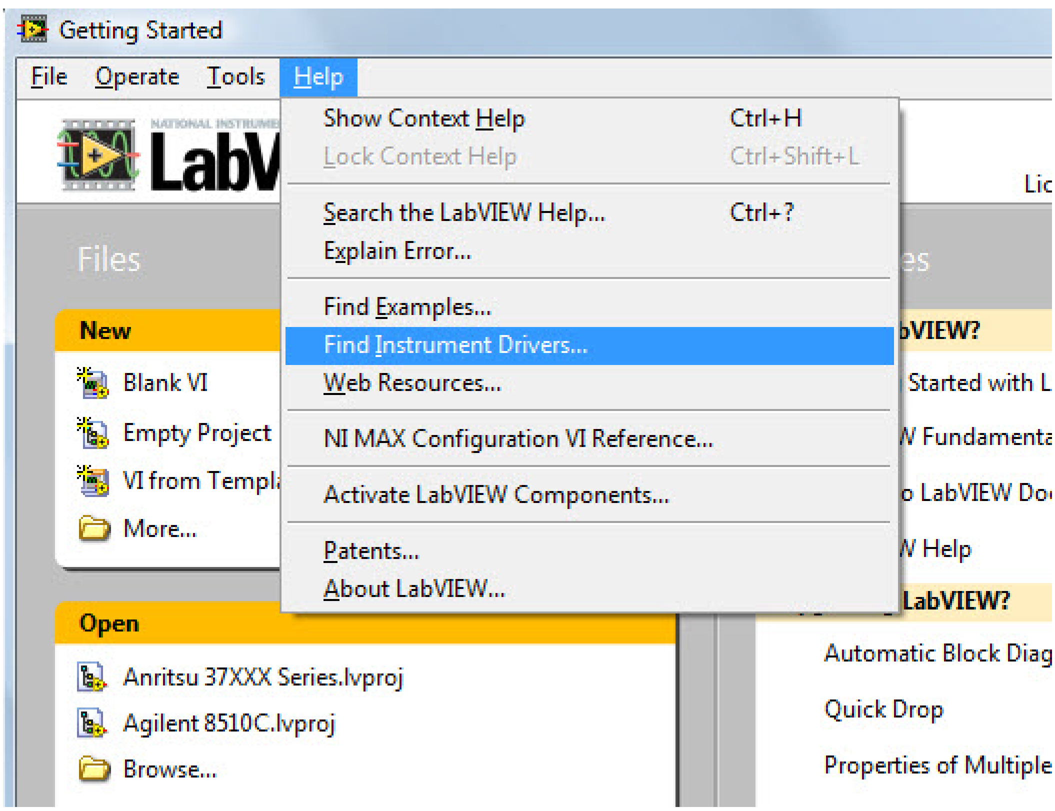
Programming Environments

Programming environments that are commonly used with test equipment include LabVIEW, LabWindows/CVI, Microsoft Visual Studio, Visual Basic 6, HP Basic, and so on.

For the examples below, we'll use LabVIEW Version 8.6 with the VectorStar MS464X LabVIEW driver.

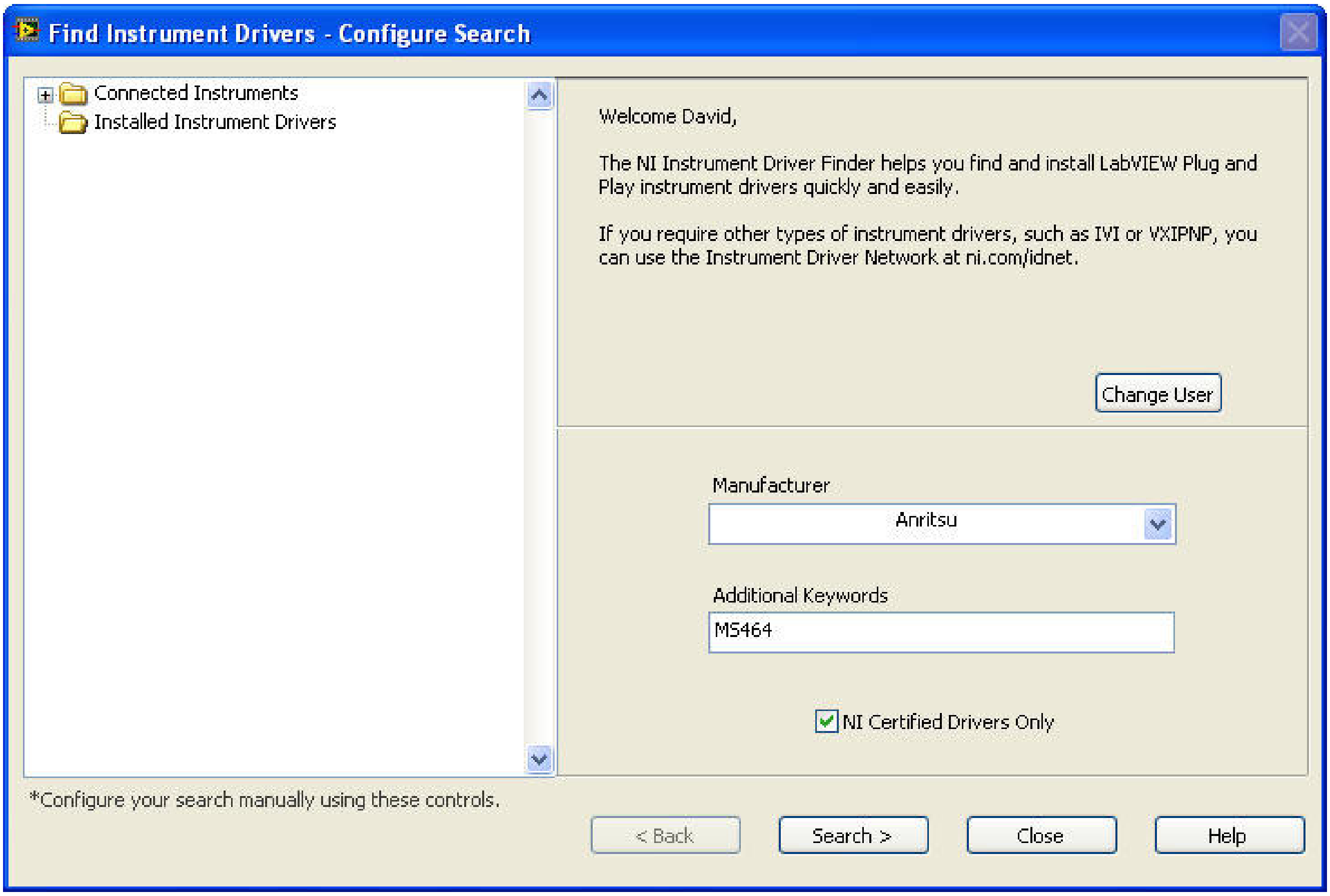
A-3 Installing the MS464X LabVIEW Driver

The VectorStar MS464X LabVIEW driver was developed and is supported by National Instruments. The driver is available at the National Instrument's Instrument Driver Network (http://www.ni.com/devzone/idnet/) or you can download it directly from within LabVIEW as shown in the following figures.



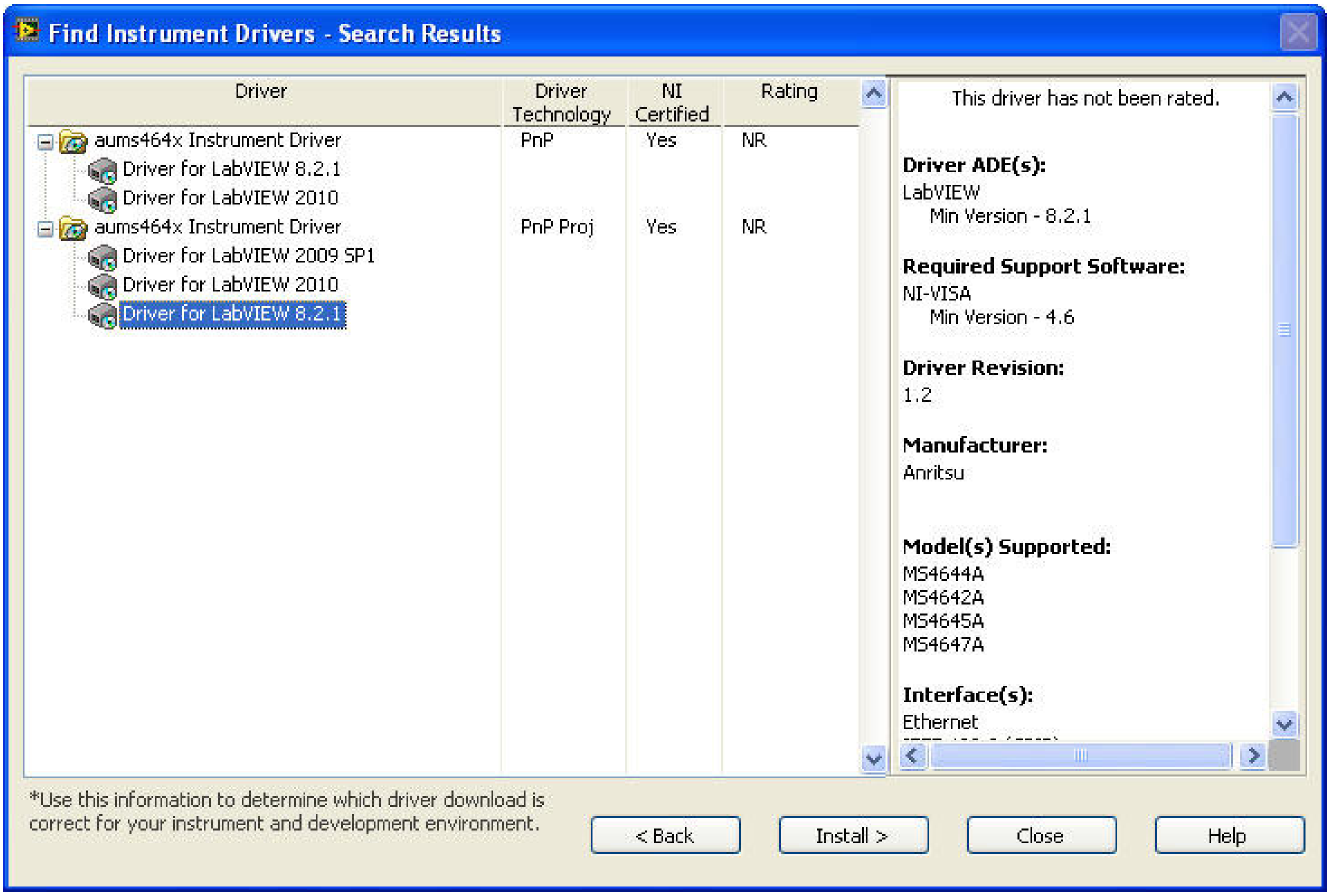
Installing instrument drivers is easy to do from the LabVIEW help menu.

Figure A-1. Installing LabVIEW Drivers



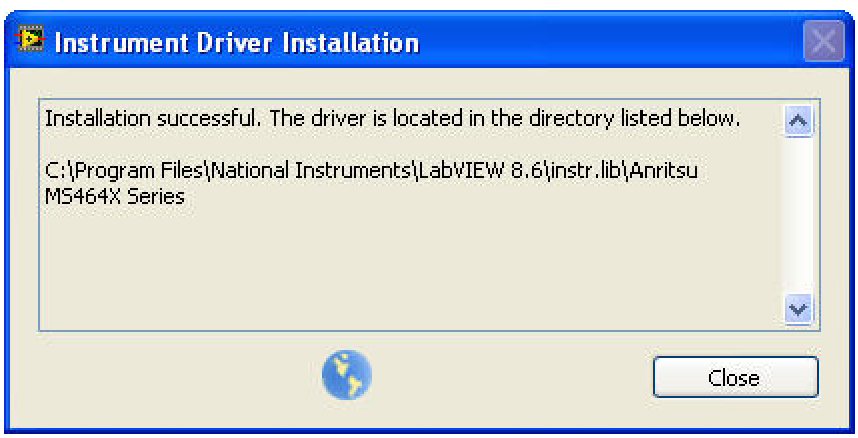
Search for Manufacturer=Anritsu and use “MS464” as the Additional Keyword. You can see on the left that no drivers are already installed.

Figure A-2. Search for Manufacturer = Anritsu + MS464



Installing the MS464X driver for LabVIEW. Select the PnP Project driver.

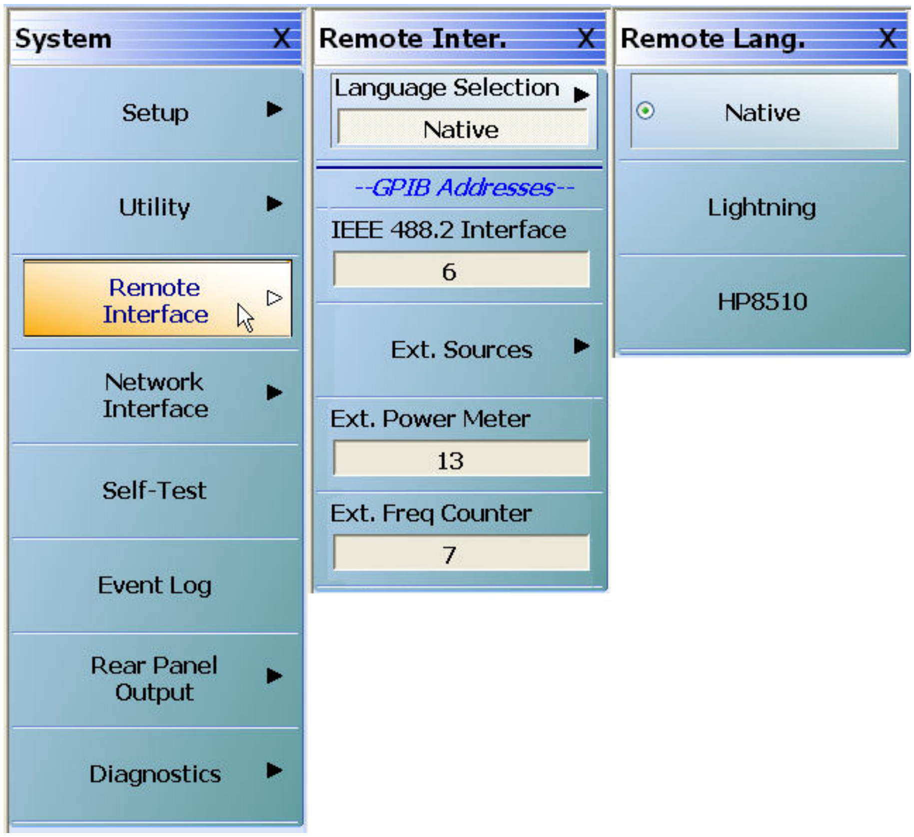
Figure A-3. Installing MS464X PnP Project Driver



It's a good idea to note where the driver has been installed.

Figure A-4. Note Driver Install Location

Set up VectorStar Remote Interface

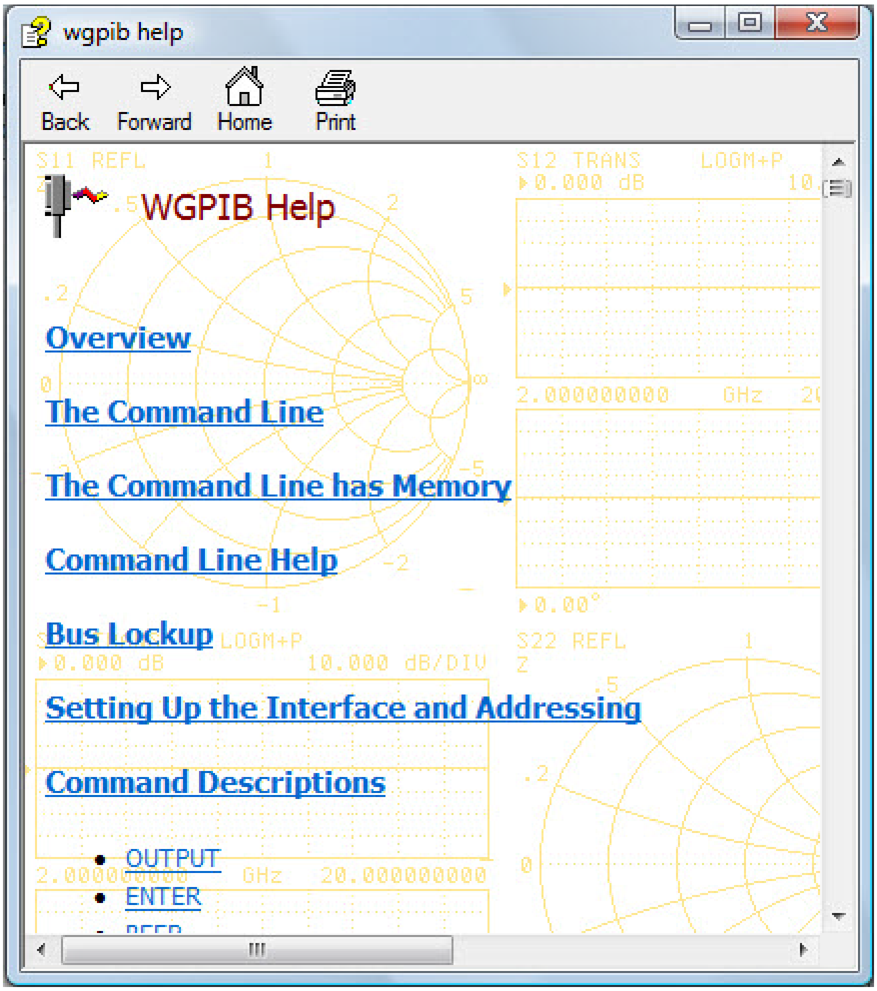


Use these menus to set up VectorStar to prioritize Native SCPI Commands for GPIB communication. We can also set this in our LabVIEW program and we'll do that in Examples 4, 7 and 8.

Figure A-5. VectorStar Remote Language Setup

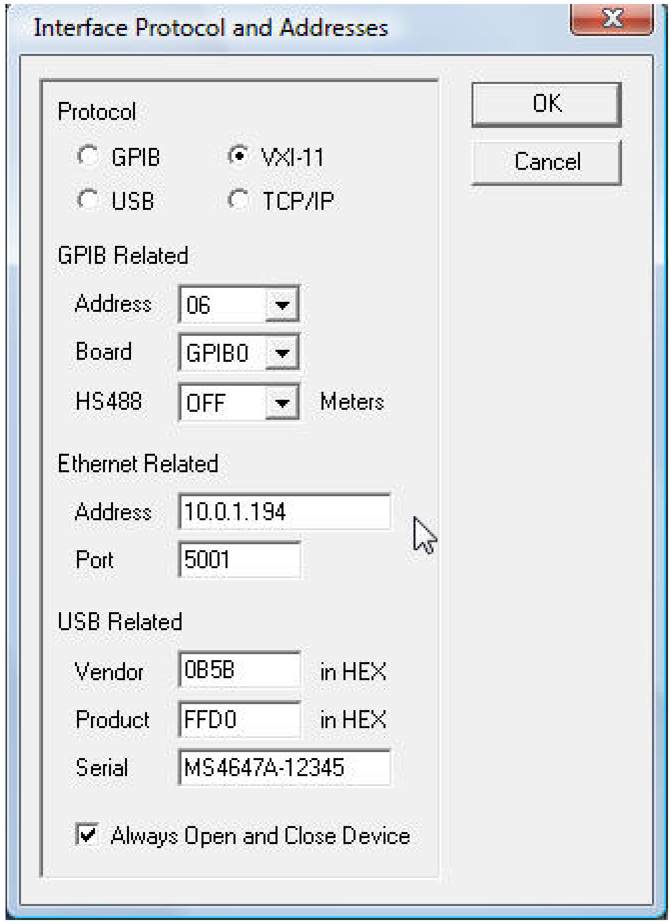
A-4 A Quick Start with the Windows GPIB Utility

Windows GPIB (WGPIB) is a useful windows program that is available from the Anritsu web site. The latest version is 3.75 and you'll want to get that one or higher to get the latest communication functions. Before writing any software, make sure you can write to and read from VectorStar over the communication protocol you're interested in.



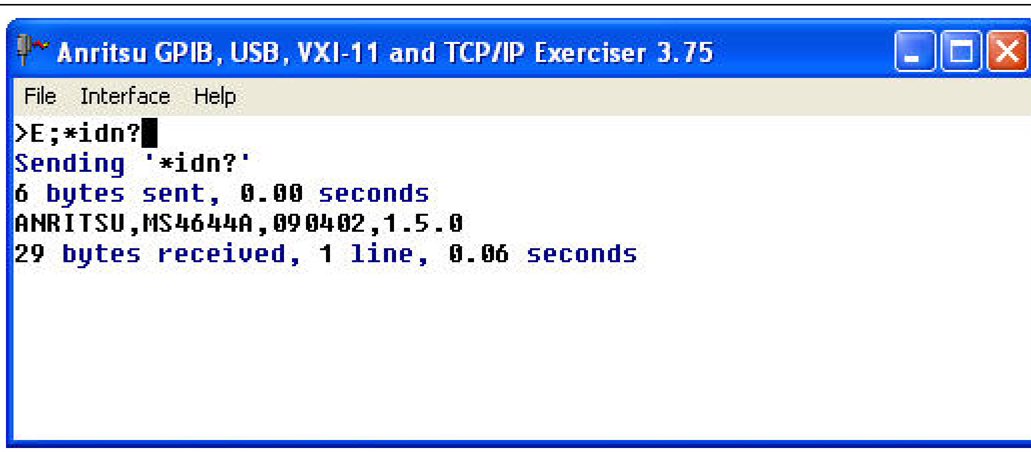
Take a quick look at WGPIB Help to get an overview of the application.

Figure A-6. WGPIB Help Home Menu



Example of setting up a VXI-11 connection to VectorStar. The VectorStar TCP/IP address is 10.0.1.194 in this example.

Figure A-7. Setup VXI-11 Connector to VectorStar VNA



The result of using WGPIB to query the instrument using the "*IDN?" command.

Figure A-8. *IDN? Query and Output Results

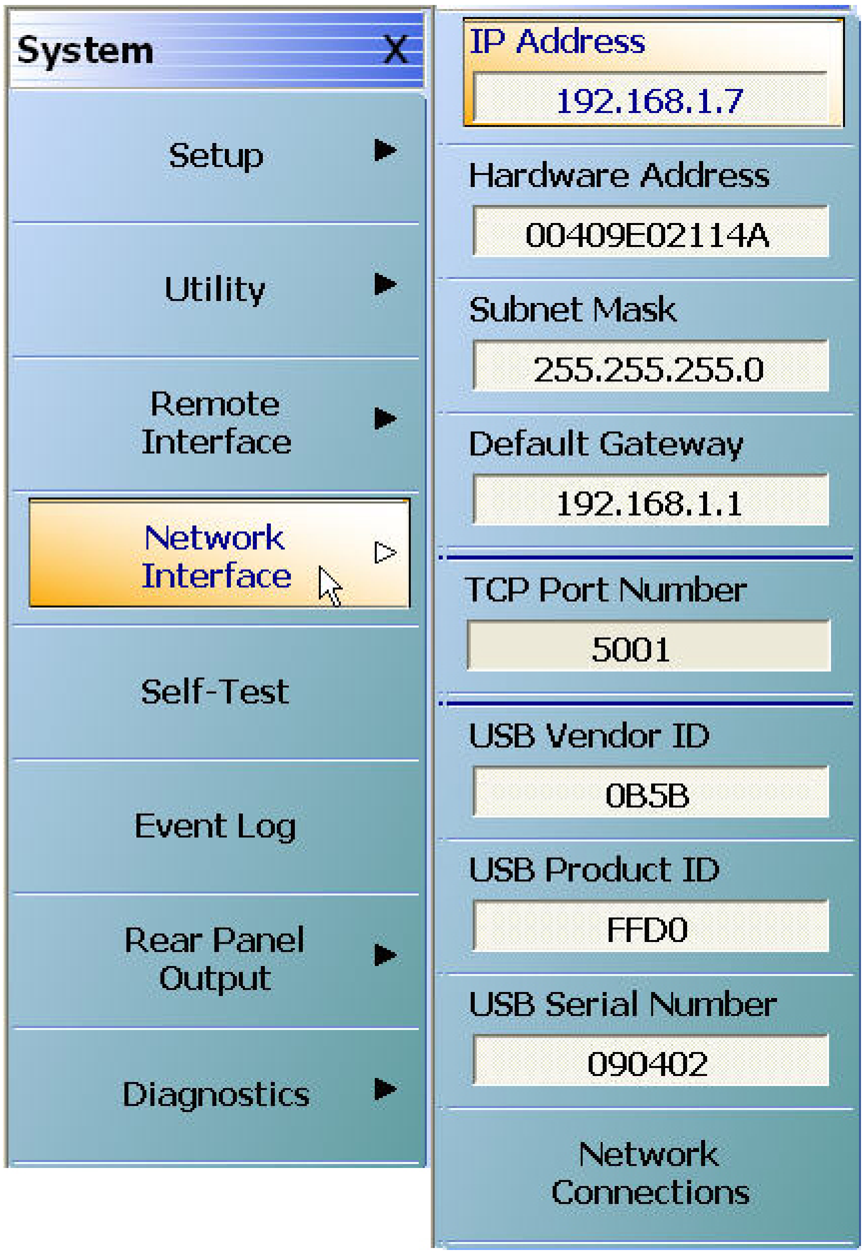
References

- LabVIEW Fundamentals Manual
- Lightning Programming Manual – 10410-00262
- VectorStar Programming Manual – 10410-00322
- VectorStar Programming Manual Supplement – 10410-00323

A-5 Programming Examples

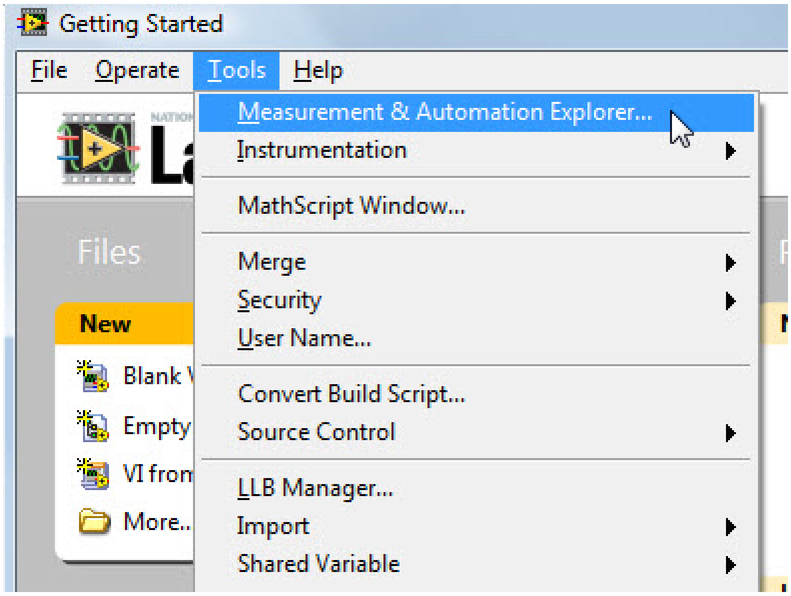
Getting Started

We'll create a few simple programs to demonstrate the use of the MS464X LabVIEW driver for controlling VectorStar.



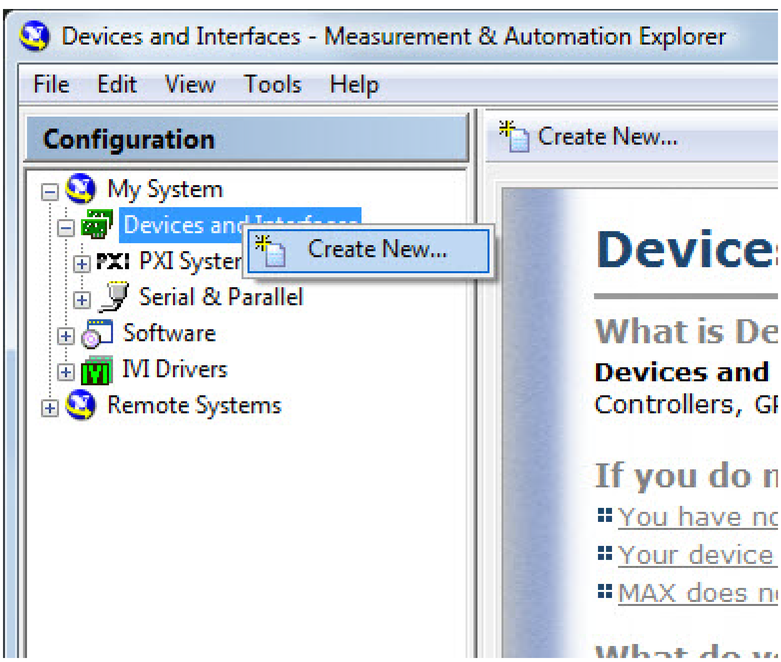
We want to communicate to VectorStar using VXI-11 (TCP/IP) so we'll need to note the IP Address of the VNA and set up a resource (a connection string).

Figure A-9. VXI-11 Communication with TCP/IP and IP Address



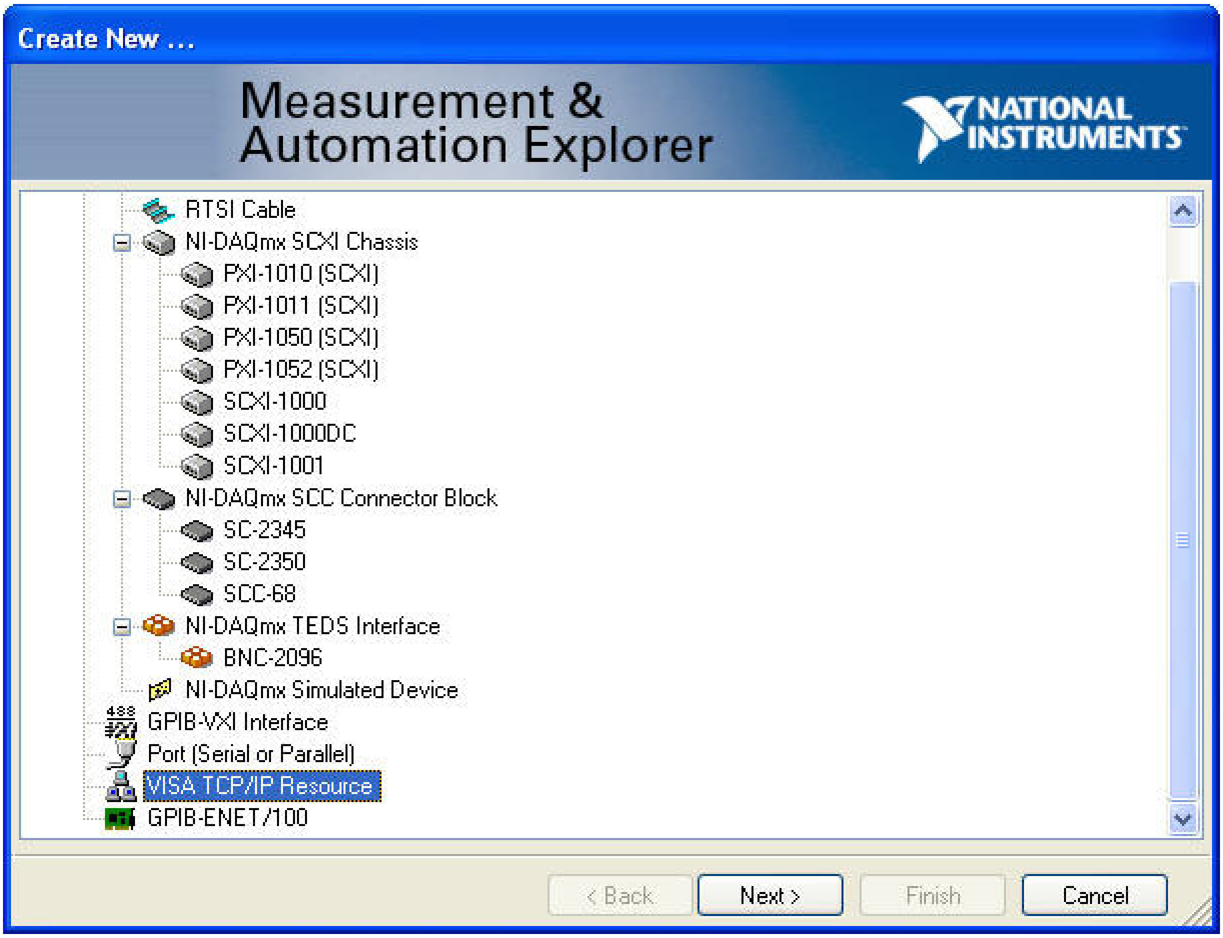
We’ll set up the VISA resource using the Measurement and Automation Explorer (MAX).

Figure A-10. VISA Resource using Measurement and Automation Explorer (MAX)



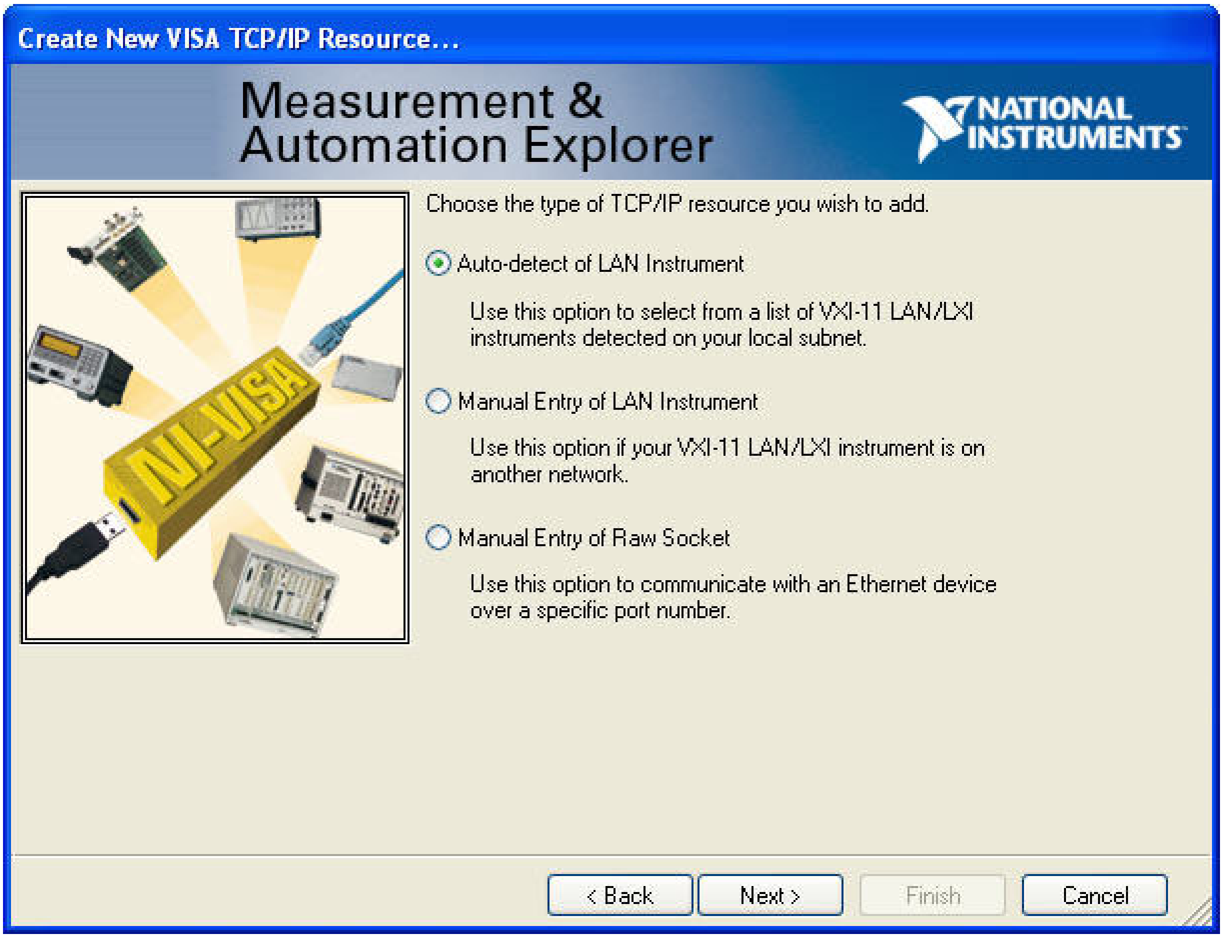
Create a new resource.

Figure A-11. MAX – Creating New Resource



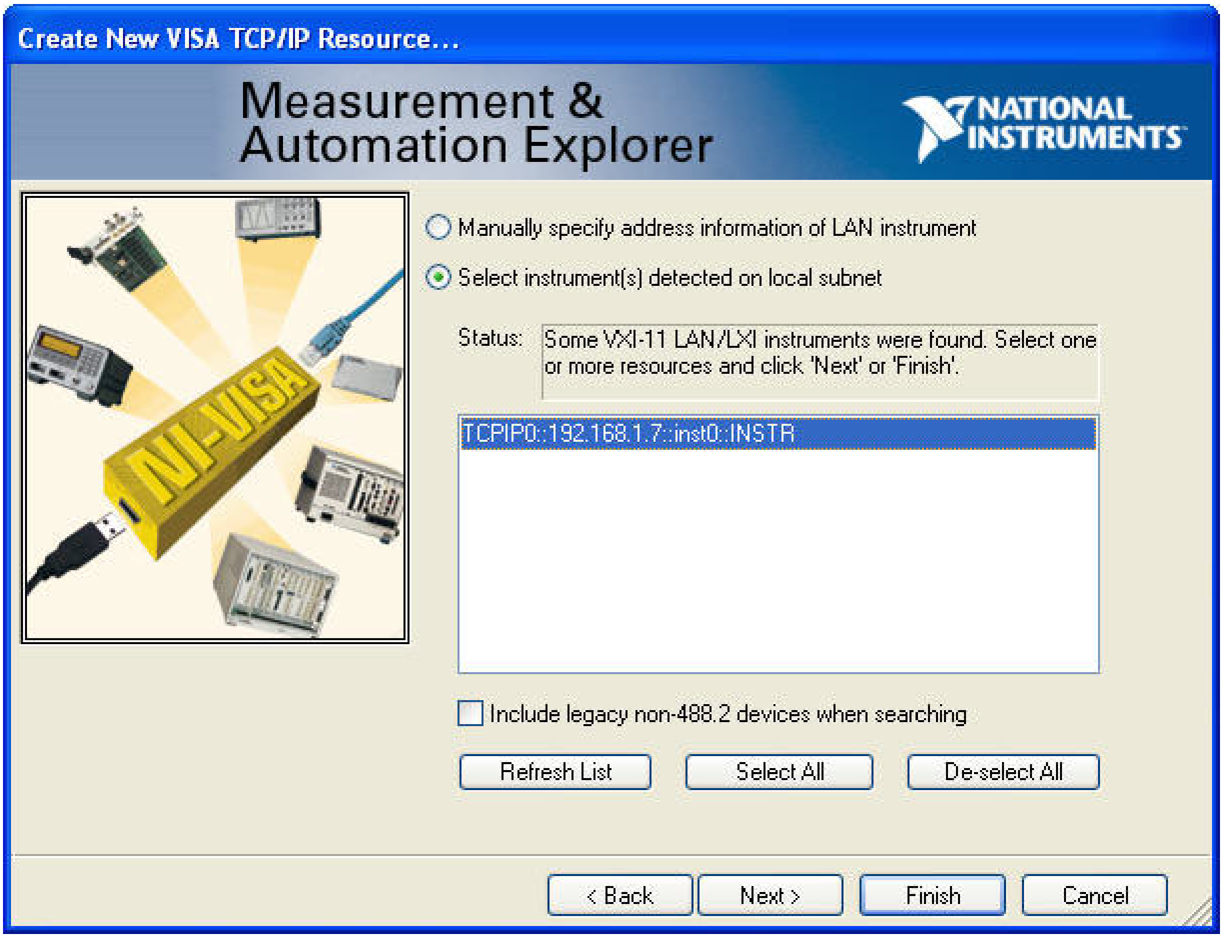
Select VISA TCP/IP Resource.

Figure A-12. MAX – Select VISA TCP/IP Resource



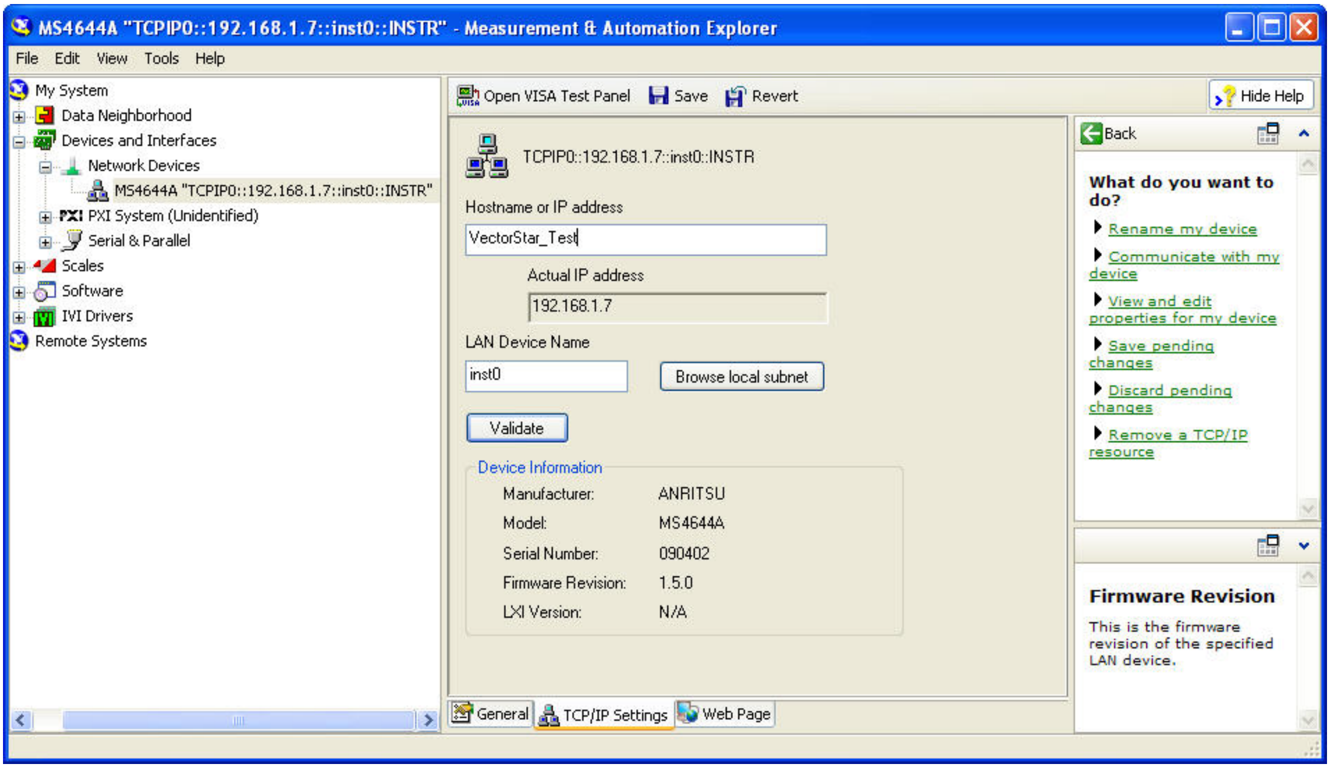
If the controlling PC and the VNA are on the same local sub-network (this is usually true if the first 3 numbers in the IP address are the same (for example, 192.168.1.x in this case), then you can probably Auto-detect the VNA.

Figure A-13. MAX – TCP/IP Auto-Detect Function



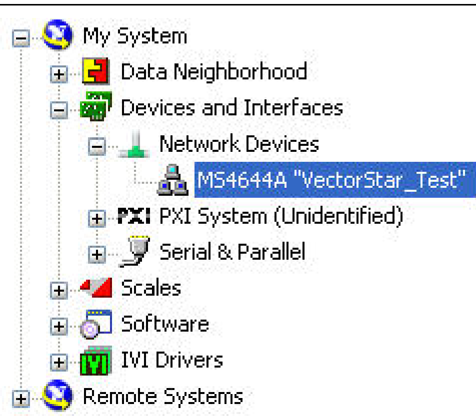
Select the detected instrument.

Figure A-14. MAX – TCP/IP Auto-Detect Function – Select Instrument



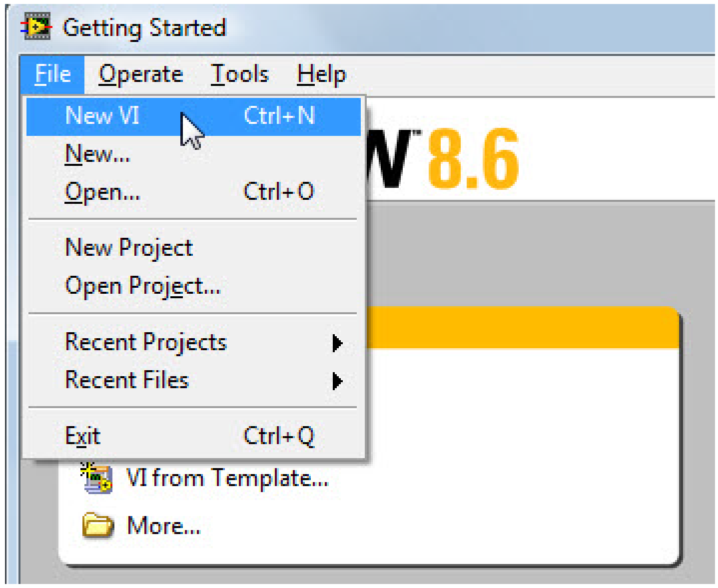
Give the instrument an alias. Later on we'll use this alias in LabVIEW.

Figure A-15. MAX – TCP/IP Auto-Detect Function – Instrument VISA Alias



Note that the VISA connection string has been replaced with the VISA alias.

Figure A-16. Instrument VISA Alias in Use

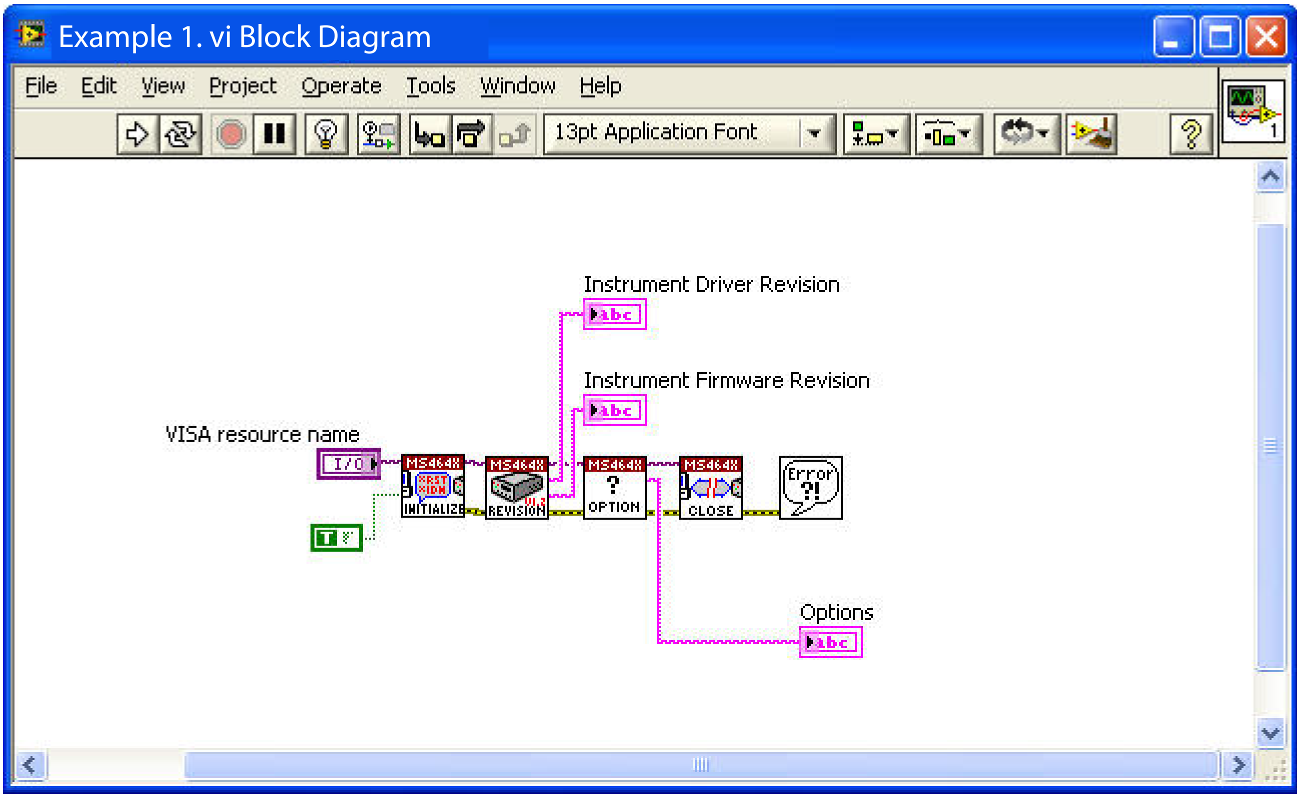


Now back in LabVIEW we're ready to start creating the first example. We'll create a new VI.

Figure A-17. LabVIEW – First Example – New VI

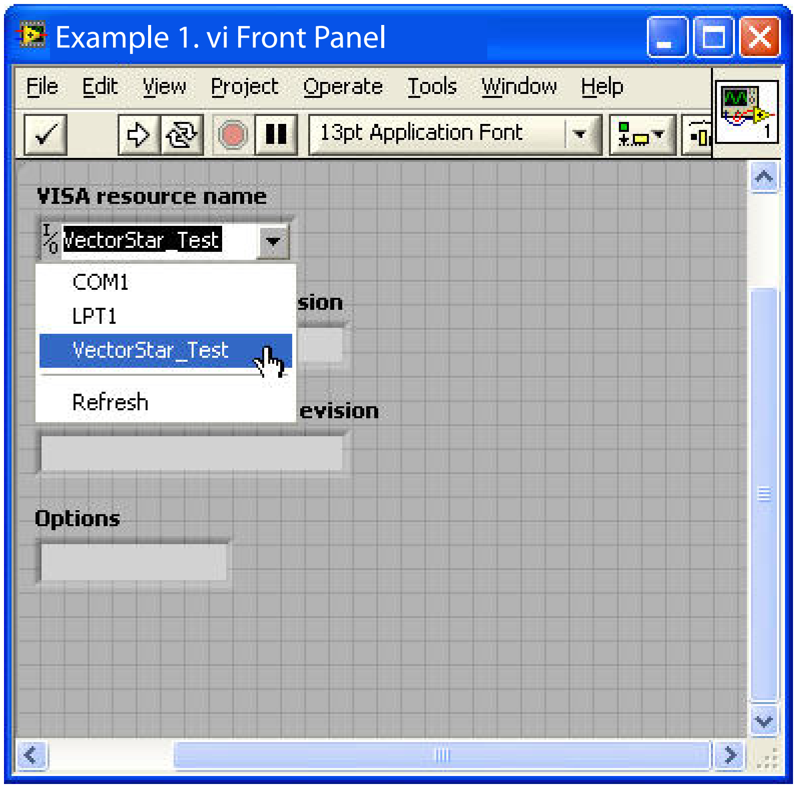
A-6 Example 1 — Open a Session to get some Instrument Information

In this first example we'll simply open a communication session to the VNA and then use two of the driver VIs to get some information about the VNA.



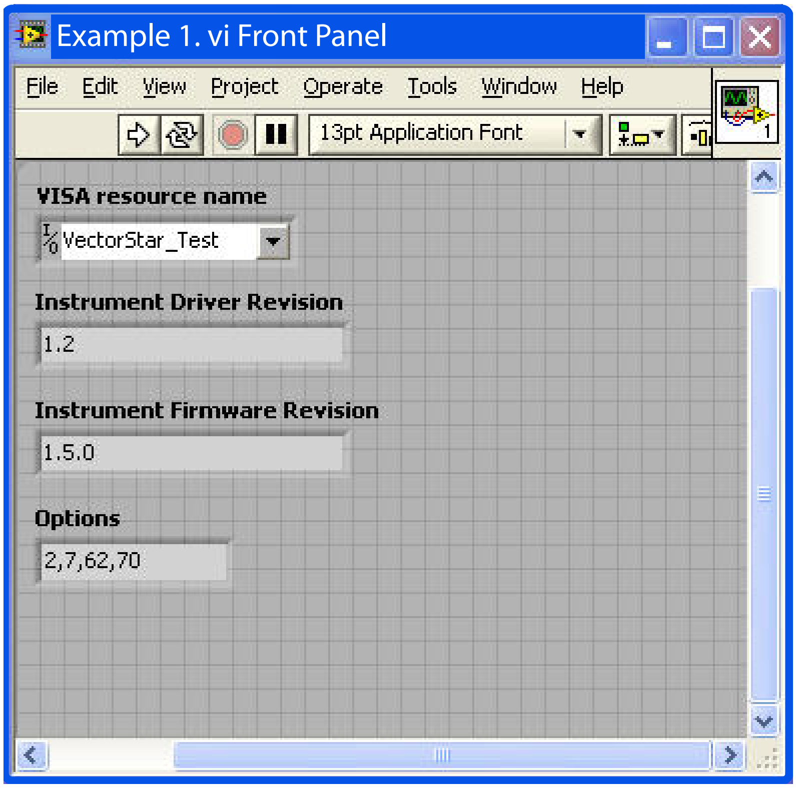
Block Diagram. Here we're using four VIs from the Anritsu MS464xB Series driver: (1) Initialize.vi, (2) Revision Query.vi, (3) Instrument Options.vi, (4) Close.vi.

Figure A-18. Example 1 – Block Diagram – Open Session – Obtain Information



Front Panel. Select “VectorStar_Test”, which was set up previously, as the VISA resource name.

Figure A-19. Example 1 – Front Panel

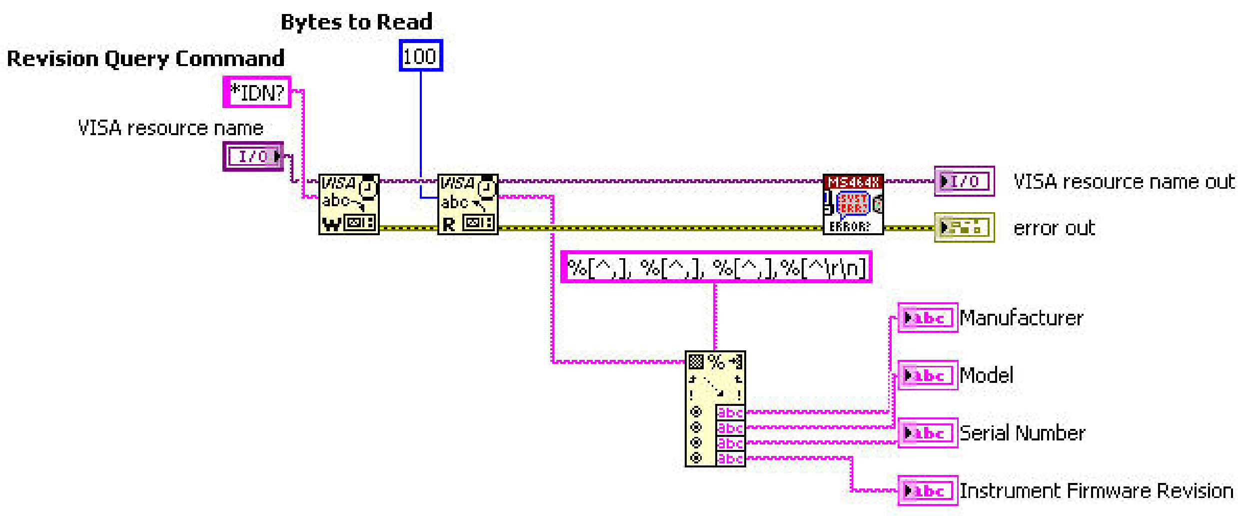


After running this VI the Instrument Driver Version, Instrument Firmware Version and Options are filled in.

Figure A-20. Example 1 – Front Panel – Fields Filled In

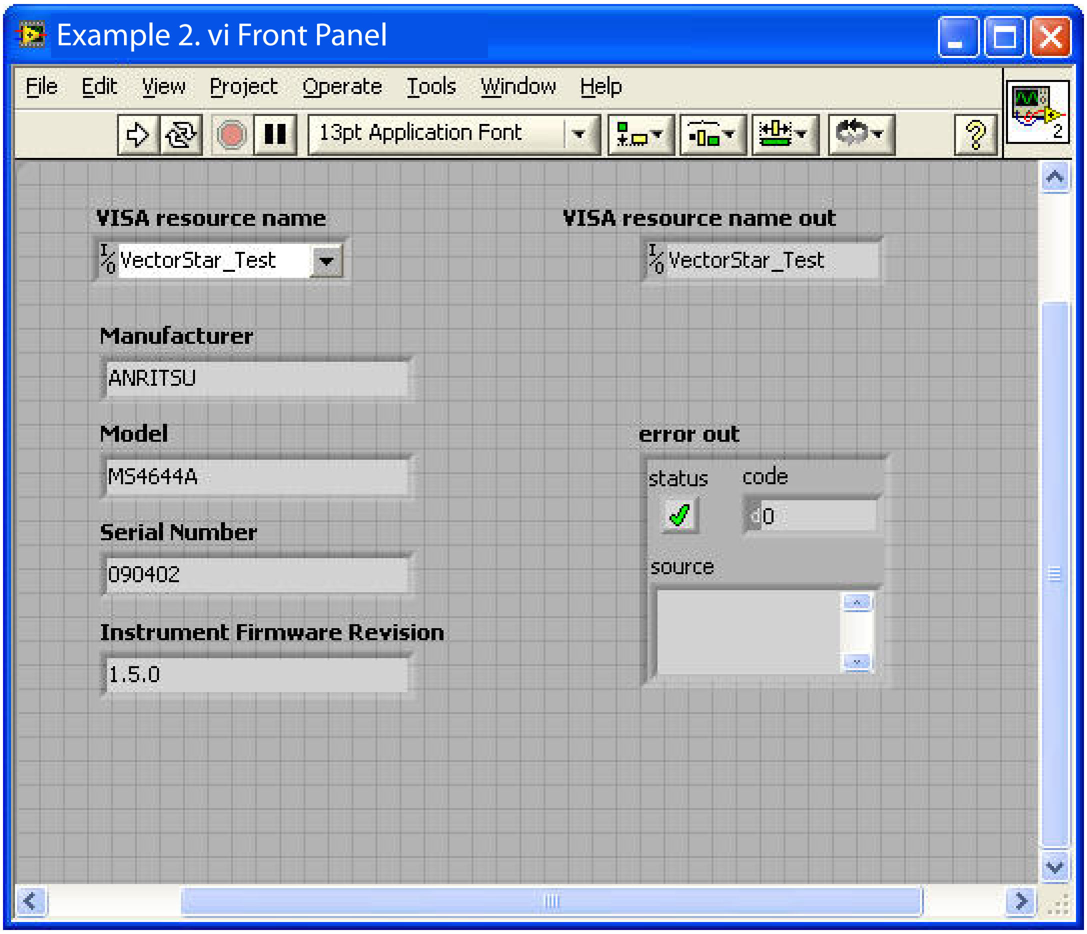
A-7 Example 2 — Sending the *IDN? Command – Display Results

The previous example used only driver VIs to get some information from the VNA. The GPIB command, “*IDN?” returns the Manufacturer, Model #, Serial Number and Firmware Version. We used this command previously in Figure 8 using the WGPIB utility. In this example we directly issue the “*IDN?” command and then parse the different parts of the response string.



Here we directly send a GPIB string using the VISA Write Function, then we read the response from the VNA using the VISA Read Function. Then we use the Scan From String Function to grab the different comma-separated values in the response string.

Figure A-21. Example 2 – Block Diagram – *IDN? Command

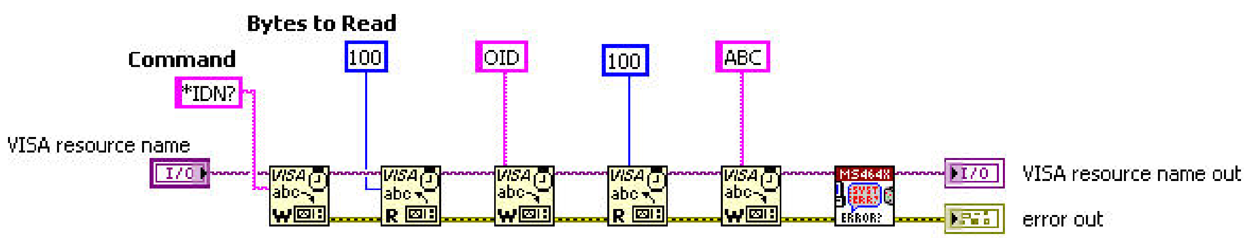


Front Panel for Example 2

Figure A-22. Example 2 – Front Panel

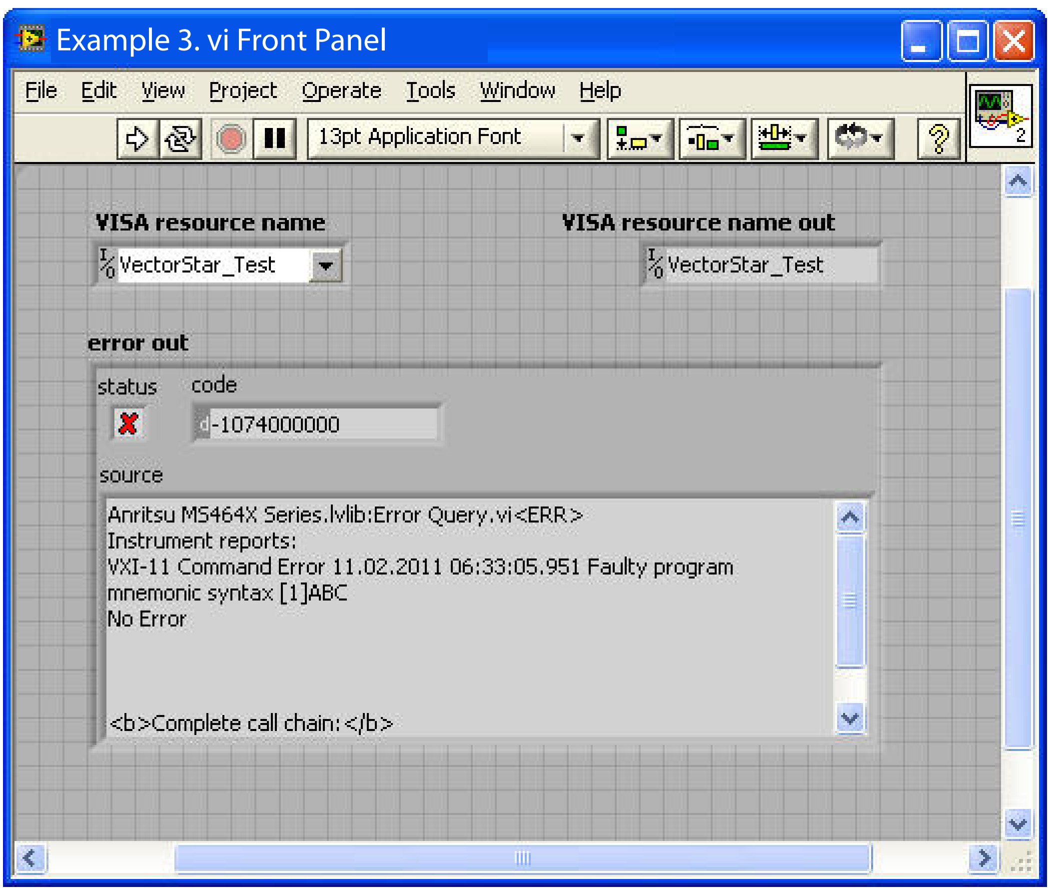
A-8 Example 3 — Error Checking

Most of the MS464X driver VIs use the Error Query.vi. This example shows that if an invalid GPIB string is sent to the VNA then the Error Query VI catches the error and displays the error message from the VNA. Here we send two valid strings: “*IDN?” and then “OID”. But the third string is not a valid GPIB command and the instrument reports this to the Front Panel.



Here we mix VISA Writes and Reads with the MS464X driver’s Error Query VI.

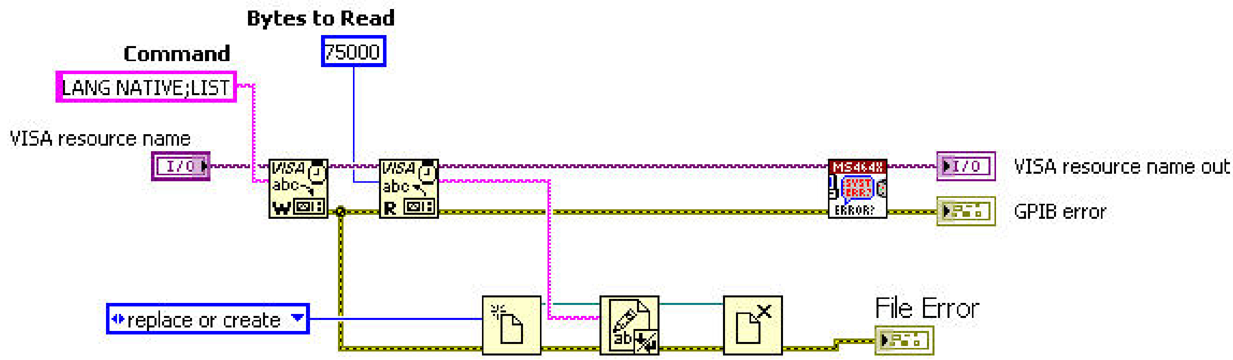
Figure A-23. Example 3 – Block Diagram – Error Checking



The “ABC” command is invalid and was captured by the Error Query VI.

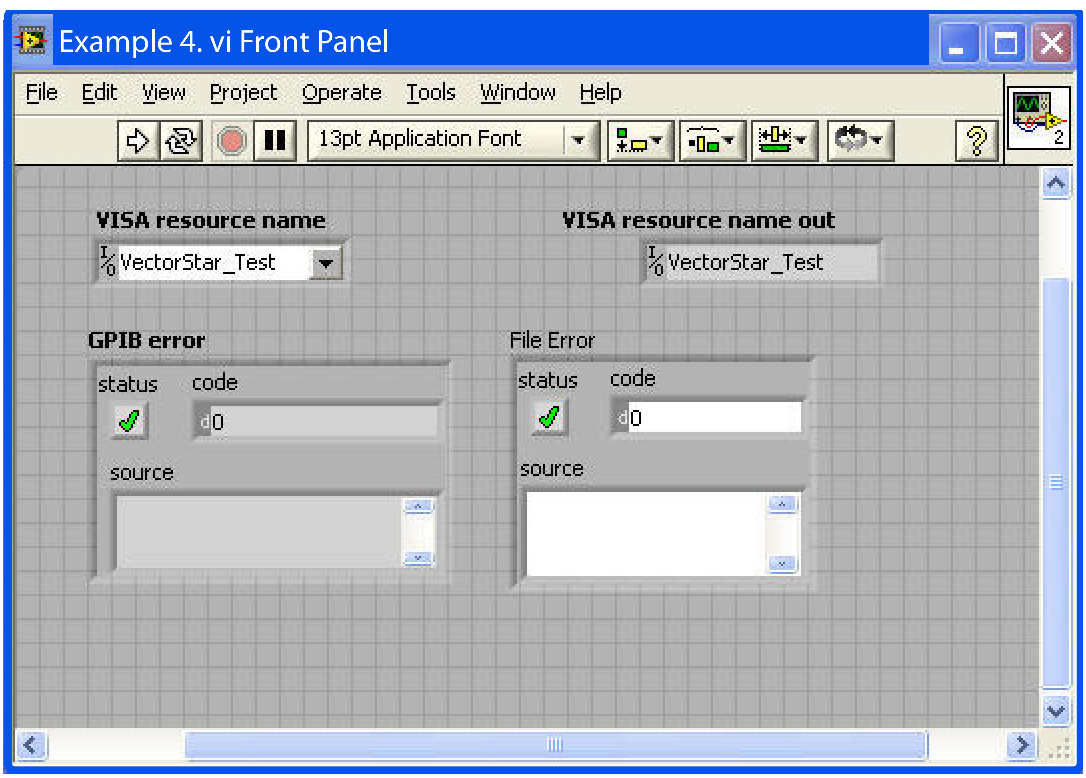
Figure A-24. Example 3 – Front Panel – Error Capture

A-9 Example 4 — LIST Command – Send to a File



Here we use VISA Write to send the LIST command to the VNA. We use “LANG NATIVE” because this will send a newline after each command in the output. If we used “LANG LIGHT” then commands are separated by a comma and the file is harder to read.

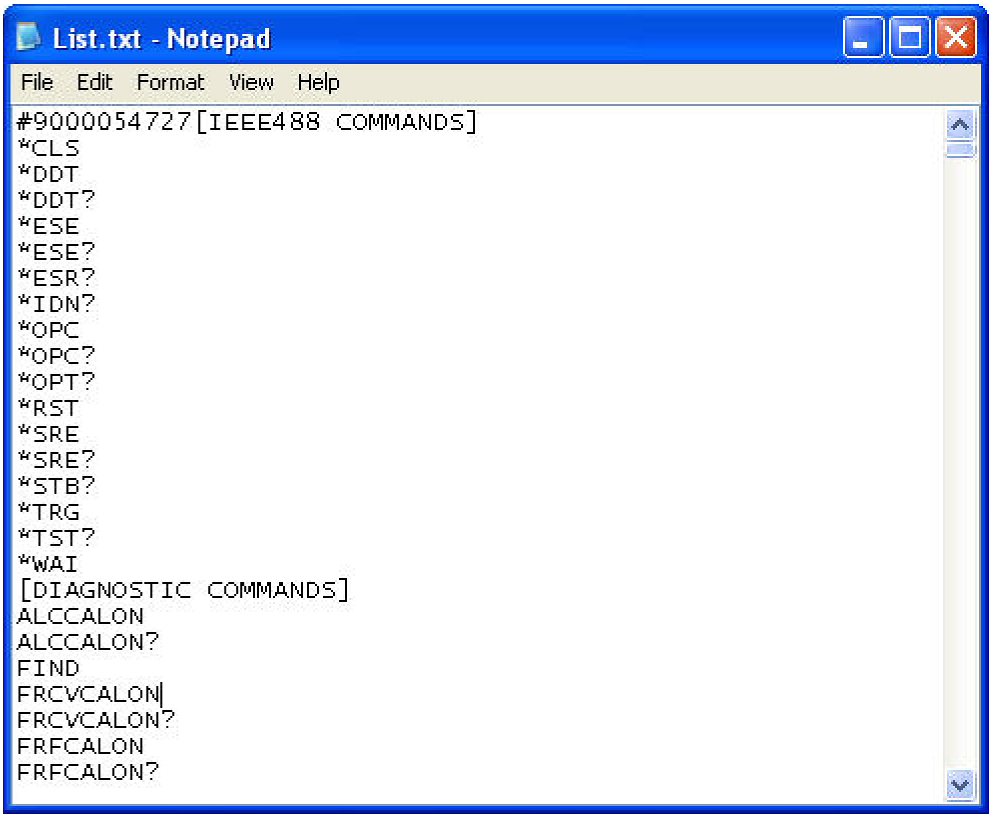
Figure A-25. Example 4 – Block Diagram – LIST Command



The Front Panel just shows us that nothing went wrong with either the GPIB code or with the File writing code.

Figure A-26. Example 4 – Front Panel

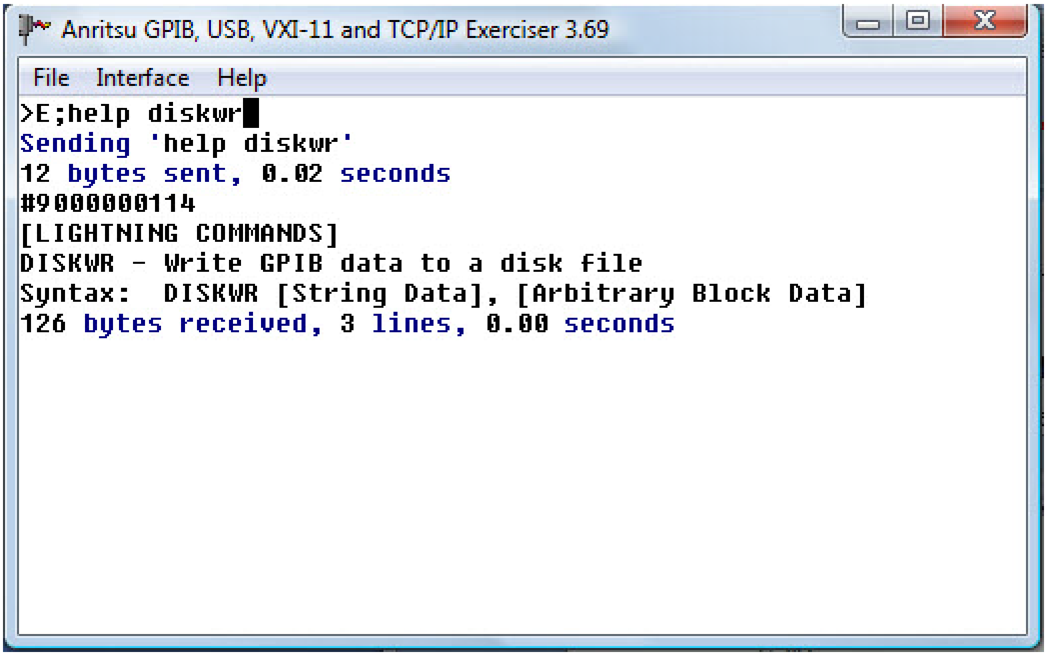
This example sends output to a file that the user may specify (a file dialog is opened).




```
#9000054727[IEEE488 COMMANDS]
*CLS
*DDT
*DDT?
*ESE
*ESE?
*ESR?
*IDN?
*OPC
*OPC?
*OPT?
*RST
*SRE
*SRE?
*STB?
*TRG
*TST?
*WAI
[DIAGNOSTIC COMMANDS]
ALCCALON
ALCCALON?
FIND
FRCVCALON
FRCVCALON?
FRFCALON
FRFCALON?
```

The List of all commands supported by VectorStar. Note that the first line has a strange “#9000054727”. This is the arbitrary block header that we will learn how to handle in Example 6. For now we’ll just ignore it.

Figure A-27. Example 4 – VectorStar VNA Supported Command List

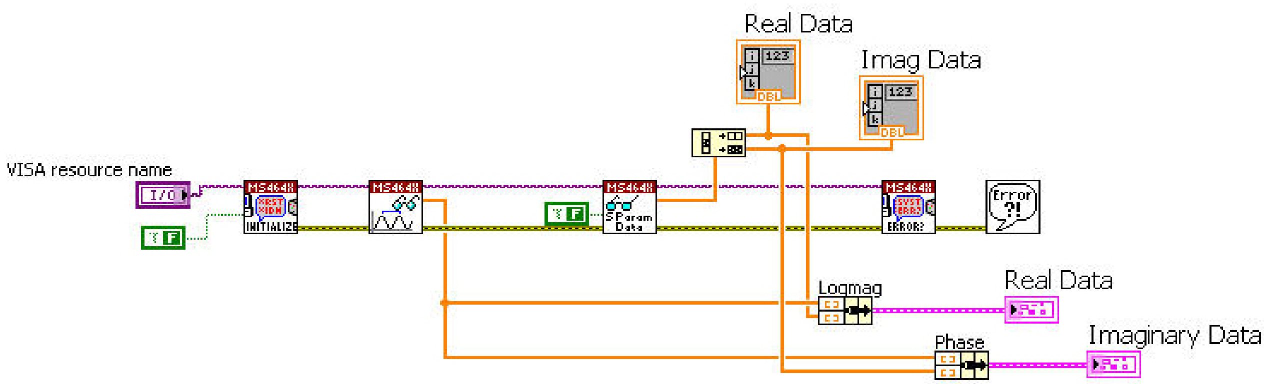


Use WGPIB to get more help on any command. Help will tell you what type of command (Lightning, Native, 8510) you're asking about and provides syntax.

Figure A-28. Example 4 – WGPIB Help for a Command

A-10 Example 5 — Acquiring Trace Data

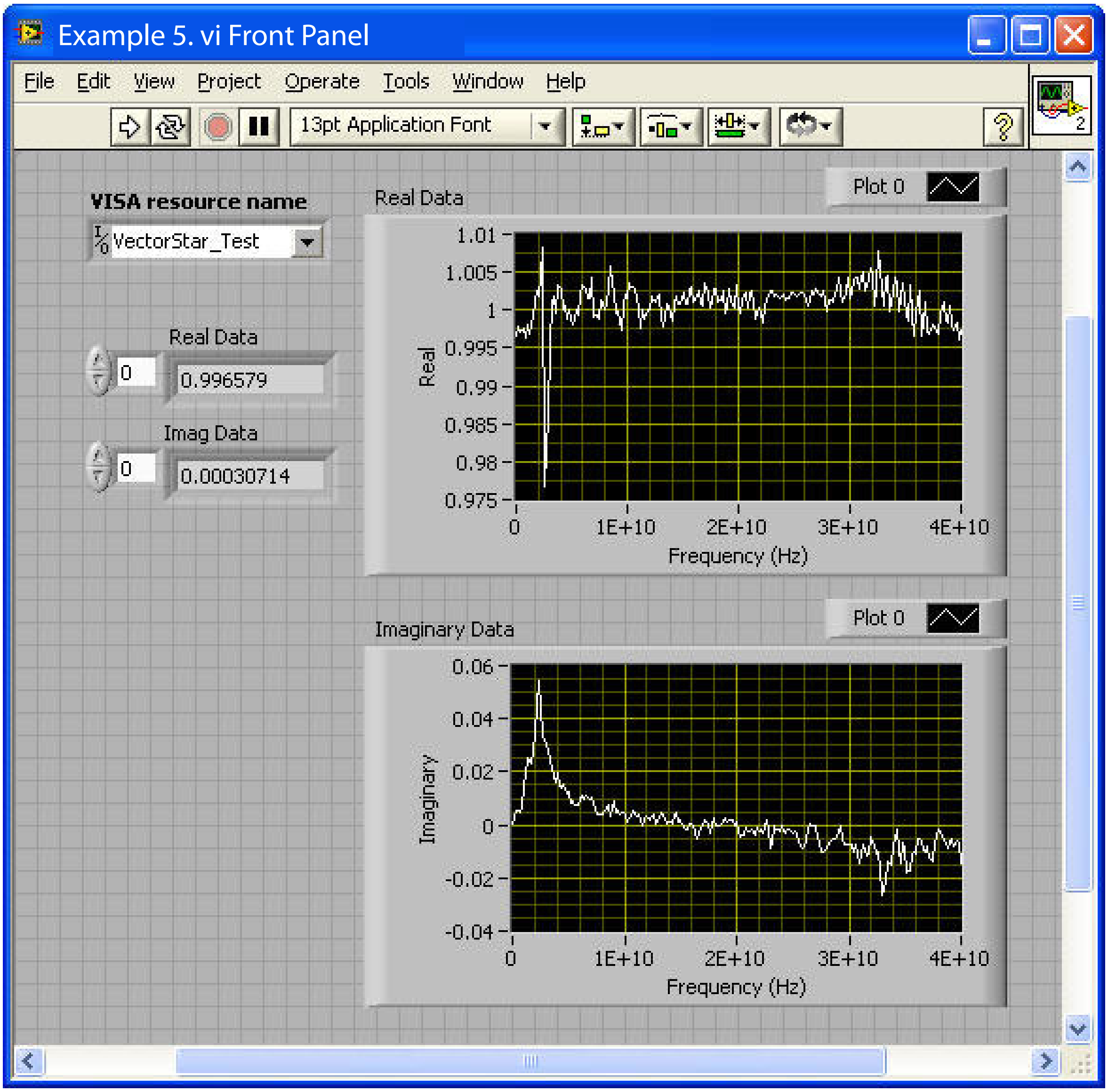
This example gets the Real/Imaginary data from the Active Trace and the data comes out in a 1 dimensional interleaved array. We need to decimate the array to get the data into two separate arrays.



We first get Frequency Values, and next we get the S-Parameter data which is Real and Imaginary.

In this example we Initialize the VNA and get Real and Imaginary Data from the Active Channel. When there are two sets of data, we need to “Decimate” the 1D array into two arrays since data 0, 2, 4, … is Real data and 1,3,5,… is Imaginary data.

Figure A-29. Example 5 – Block Diagram – Acquire Trace Data



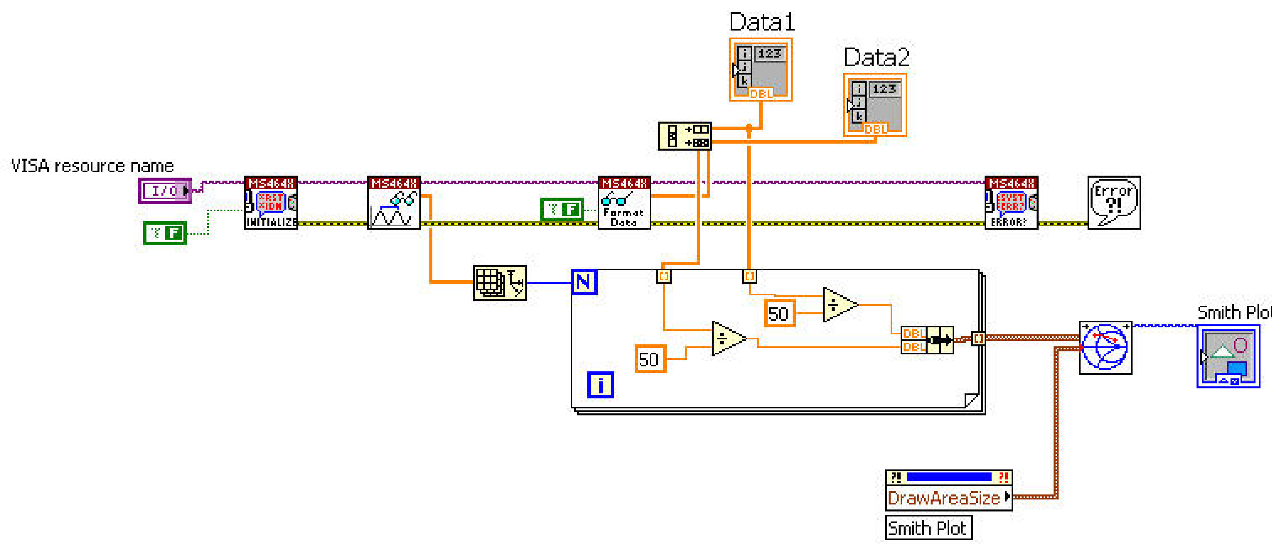
We send separate traces to each XY Graph.

Figure A-30. Example 5 – Front Panel and Resultant XY Graphs

A-11 Example 6 – Smith Chart Data

This example is similar to the previous example except here we need to make sure that the active channel is set to Smith Chart (or do an optional instrument reset which sets up smith charts on trace 1 and 4). Although we don't need the frequency data to display the smith chart, we use the Read Frequency Values VI to get the frequency list and then grab the data count from the number of frequencies. This data count is passed to the For Loop control.

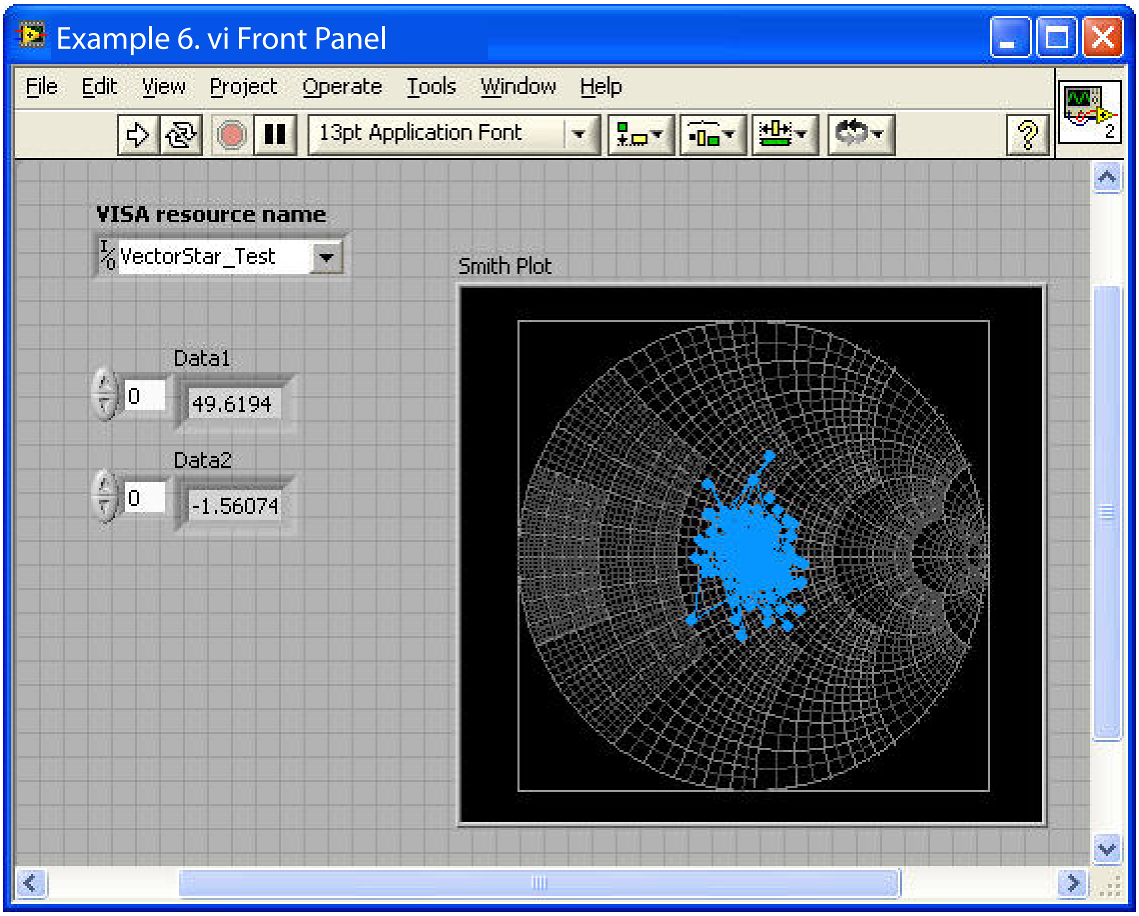
By default Trace1 is set to output impedance values. Smith chart actually takes a normalized impedance (normalized to 1) so we divide the impedance/reactance pairs by 50 (ohms) to get normalized smith chart data.



We again use Read Frequency Values, but here we get the Formatted Data for the active channel (which is a Smith Chart).

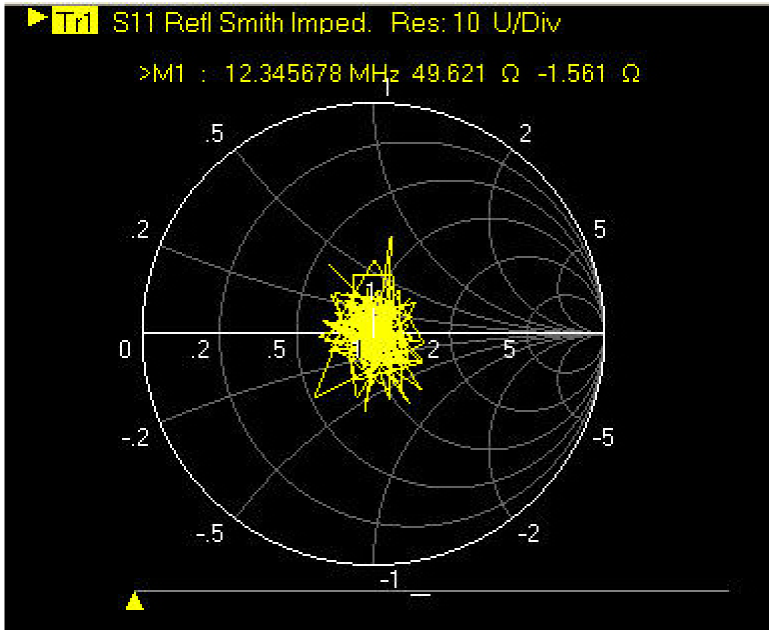
Use the Frequency Array to get the number of points and feed that into the For Loop control. The Channel Data from the Active Trace is Impedance/Reactance so we divide by Z_0 (or 50 ohms, hardcoded here) to get the smith chart data.

Figure A-31. Example 6 – Block Diagram – Smith Chart



Front Panel Display.

Figure A-32. Example 6 – Front Panel Display and Resultant Smith Chart

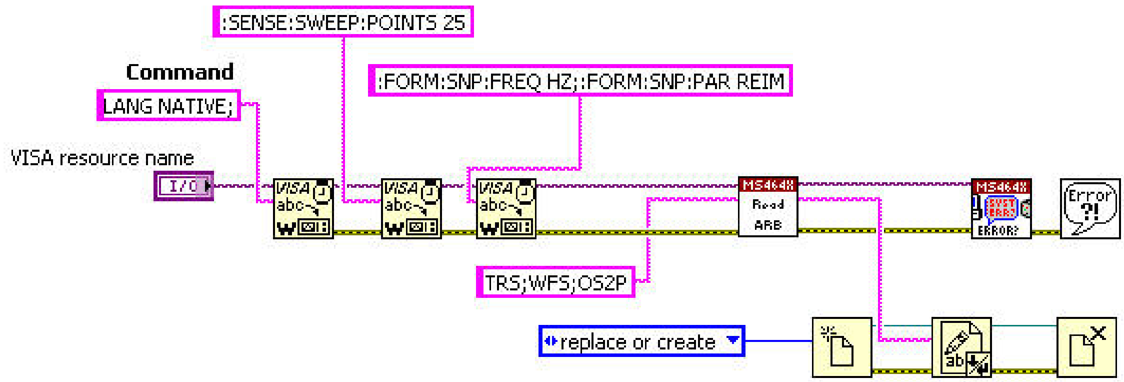


Here's the display on VectorStar.

Figure A-33. Example 6 – VectorStar Smith Chart Trace 1 Display

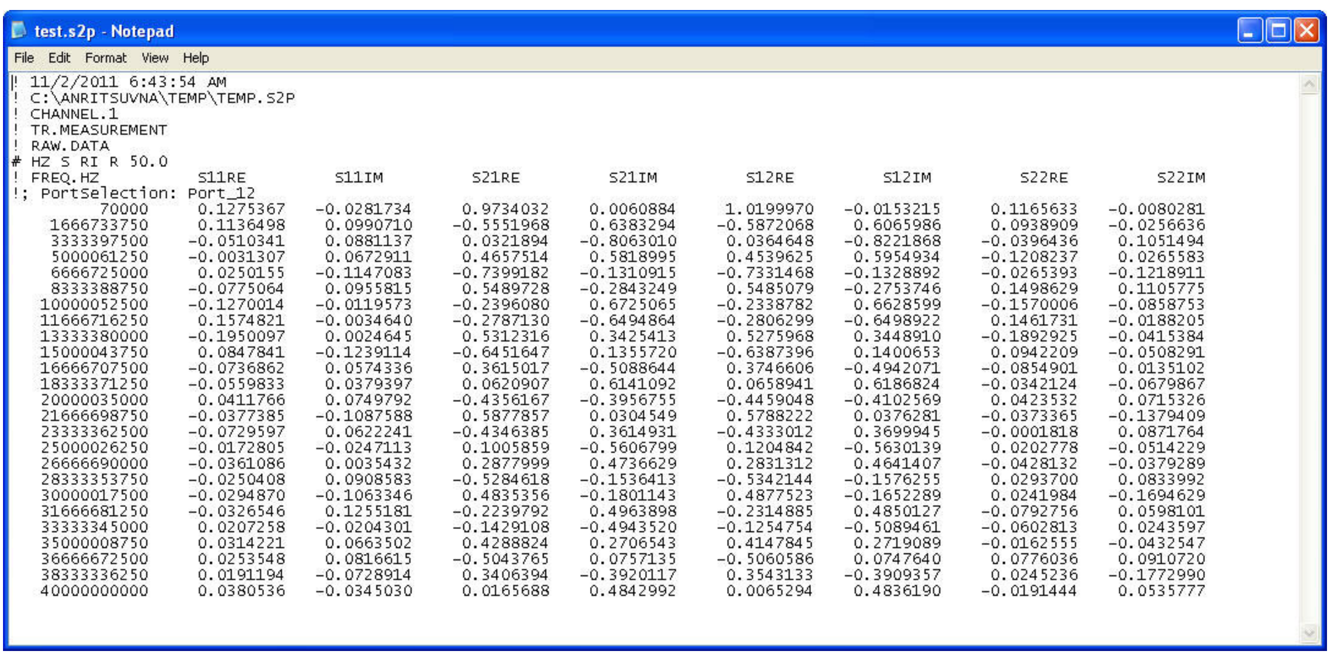
A-12 Example 7 – Output S2P File

This is a good example that uses some VISA Writes and some MS464X driver VIs to accomplish some useful things. Specifically, we are going to send some Native VectorStar commands along with some Lightning commands to output an S2P file from the VNA to the PC.



Here we use a combination of VISA and driver commands. First Native GPIB strings are sent to setup the S2P output to 25 data points, Frequency in Hz, and Data Format to Real/Imaginary. I used the Read ASCII ARB function to strip off the header.

Figure A-34. Example 7 – Block Diagram – Output S2P File



test.s2p - Notepad

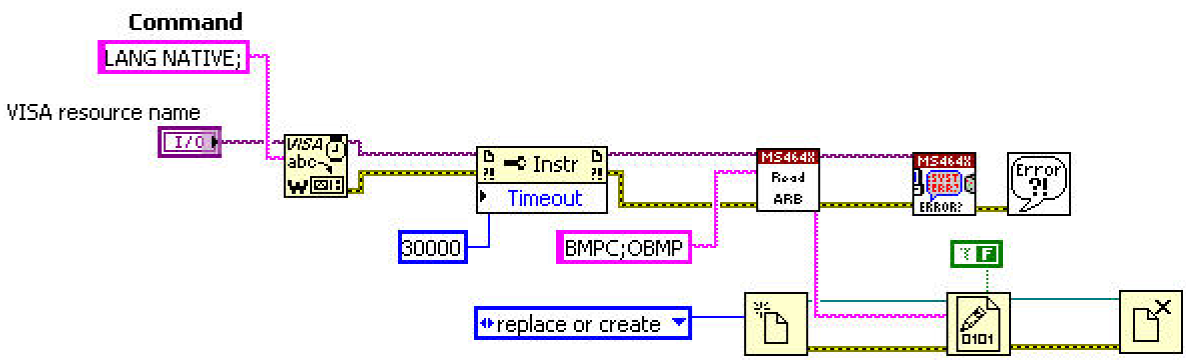
File Edit Format View Help

```
! 11/2/2011 6:43:54 AM
! C:\ANRITSUVNA\TEMP\TEMP.S2P
! CHANNEL.1
! TR.MEASUREMENT
! RAW.DATA
# HZ S RI R 50.0
! FREQ.HZ        S11RE      S11IM      S21RE      S21IM      S12RE      S12IM      S22RE      S22IM
!; PortSelection: Port_12
      70000   0.1275367 -0.0281734  0.9734032  0.0060884  1.0199970 -0.0153215  0.1165633 -0.0080281
 1666733750   0.1136498  0.0990710 -0.5551968  0.6383294 -0.5872068  0.6065986  0.0938909 -0.0256636
 3333397500  -0.0510341  0.0881137  0.0321894 -0.8063010  0.0364648 -0.8221868 -0.0396436  0.1051494
 5000061250  -0.0031307  0.0672911  0.4657514  0.5818995  0.4539625  0.5954934 -0.1208237  0.0265583
 6666725000   0.0250155 -0.1147083 -0.7399182 -0.1310915 -0.7331468 -0.1328892 -0.0265393 -0.1218911
 8333388750  -0.0775064  0.0955815  0.5489728 -0.2843249  0.5485079 -0.2753746  0.1498629  0.1105775
10000052500  -0.1270014 -0.0119573 -0.2396080  0.6725065 -0.2338782  0.6628599 -0.1570006 -0.0858753
11666716250   0.1574821 -0.0034640 -0.2787130 -0.6494864 -0.2806299 -0.6498922  0.1461731 -0.0188205
13333380000  -0.1950097  0.0024645  0.5312316  0.3425413  0.5275968  0.3448910 -0.1892925 -0.0415384
15000043750   0.0847841 -0.1239114 -0.6451647  0.1355720 -0.6387396  0.1400653  0.0942209 -0.0508291
16666707500  -0.0736862  0.0574336  0.3615017 -0.5088644  0.3746606 -0.4942071 -0.0854901  0.0135102
18333371250  -0.0559833  0.0379397  0.0620907  0.6141092  0.0658941  0.6186824 -0.0342124 -0.0679867
20000035000   0.0411766  0.0749792 -0.4356167 -0.3956755 -0.4459048 -0.4102569  0.0423532  0.0715326
21666698750  -0.0377385 -0.1087588  0.5877857  0.0304549  0.5788222  0.0376281 -0.0373365 -0.1379409
23333362500  -0.0729597  0.0622241 -0.4346385  0.3614931 -0.4333012  0.3699945 -0.0001818  0.0871764
25000026250  -0.0172805 -0.0247113  0.1005859 -0.5606799  0.1204842 -0.5630139  0.0202778 -0.0514229
26666690000  -0.0361086  0.0035432  0.2877999  0.4736629  0.2831312  0.4641407 -0.0428132 -0.0379289
28333353750  -0.0250408  0.0908583 -0.5284618 -0.1536413 -0.5342144 -0.1576255  0.0293700  0.0833992
30000017500  -0.0294870 -0.1063346  0.4835356 -0.1801143  0.4877523 -0.1652289  0.0241984 -0.1694629
31666681250  -0.0326546  0.1255181 -0.2239792  0.4963898 -0.2314885  0.4850127 -0.0792756  0.0598101
33333345000   0.0207258 -0.0204301 -0.1429108 -0.4943520 -0.1254754 -0.5089461 -0.0602813  0.0243597
35000008750   0.0314221  0.0663502  0.4288824  0.2706543  0.4147845  0.2719089 -0.0162555 -0.0432547
36666672500   0.0253548  0.0816615 -0.5043765  0.0757135 -0.5060586  0.0747640  0.0776036  0.0910720
38333336250   0.0191194 -0.0728914  0.3406394 -0.3920117  0.3543133 -0.3909357  0.0245236 -0.1772990
40000000000   0.0380536 -0.0345030  0.0165688  0.4842992  0.0065294  0.4836190 -0.0191444  0.0535777
```

Transfer of an S2P file to the pc. When the VI runs it puts up a dialog to allow the user to select a file name. Make sure to save the file with an ".S2P" extension.

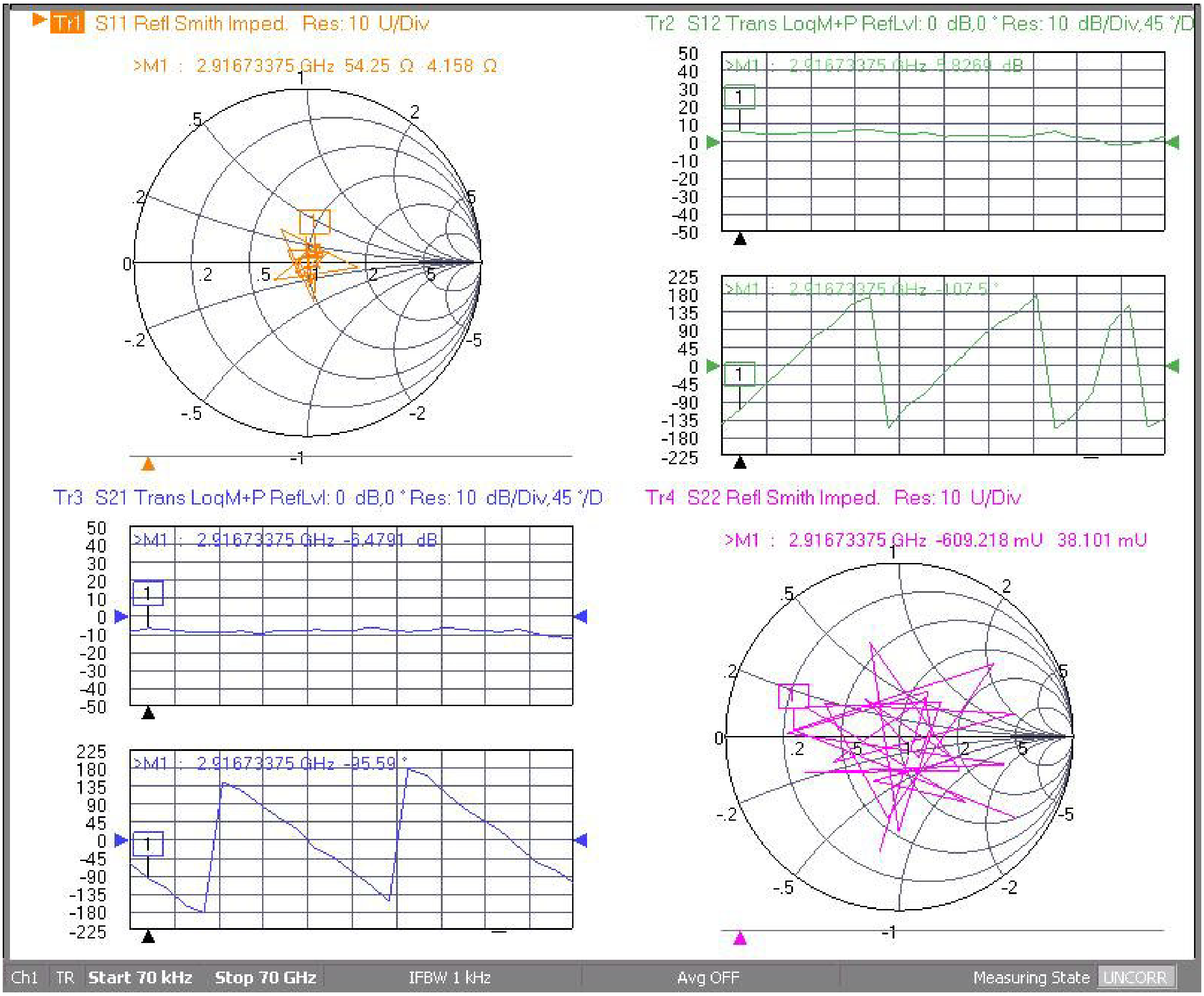
Figure A-35. Example 7 – S2P File Transfered to the PC Controller

A-13 Example 8 – Output BMP File



We can use a similar technique to get the bitmap data to a file. Here we use the Lightning commands "BMPC;OBMP" to output a bitmap file. BMPC selects color on white as the color scheme – this makes for better printouts. The "Read ARB.vi" again is used to strip off the arbitrary block header and place the bitmap data into a file. If we didn't strip off the header the bitmap file would be corrupt. Note that we write the file to a binary file.

Figure A-36. Example 8 – Block Diagram – Output BMP File



When the VI runs it puts up a dialog to allow the user to select a file name. Make sure to save the file with a ".bmp" extension.

Figure A-37. Example 8 – Front Display Graphic Output as a BMP File

Appendix B — Programming with LabWindows/CVI

B-1 Introduction

This document provides an overview of programming techniques for controlling the VectorStar VNA using the au464X LabWindows/CVI driver over GPIB, TCP/IP (using VXI-11), and USB. This document assumes some previous knowledge of LabWindows/CVI. (Note that we often use the term CVI in this document to refer to LabWindows/CVI.)

B-2 Overview

Programming Basics

There are many cases where a user may want programmatic control of the VNA. Examples include automating a test sequence, manufacturing testing, orchestrating a complex measurement involving various pieces of test equipment, gathering a time series of data, or as a convenient way of getting data, files or images transferred from the VNA to a pc for further analysis.

GPIB Mnemonics

VectorStar has a fairly large set of GPIB commands. This includes a set of Native commands, Lightning commands and Agilent 8510 commands. You use the same commands regardless of the communication method employed. See Example 4 for using the LIST command to output the full set of supported GPIB mnemonics directly from the VNA. Because VectorStar supports Lightning commands we can use the VectorStar au464x LabWindows/CVI driver to control VectorStar but also send some Lightning commands if necessary. This document uses a combination of au464x driver functions and VISA Reads and Writes to send commands to the VNA and to get data from the VNA. With this combination method, we use the driver to accomplish many simple activities while the VISA commands allow a full range of command including VectorStar Native commands (see example 6). Thus the full set of VectorStar functionality is available to the LabWindows/CVI programmer.

What is VISA?

VISA (Virtual Instrument System Architecture) is an I/O software standard for communicating with test instruments like VectorStar over any of the bus architectures which VectorStar supports. A VISA driver is available from both National Instruments and Agilent. National Instruments VISA drivers are available for the following operating systems: Windows (all versions), Mac OS X, Linux, Mandriva, RedHat, SUSE, PharLap, VxWorks, Pocket PC 2003, and Windows CE/Mobile. It’s always a good idea to get the latest driver (Version 5.1.1). If you plan to also do lots of VISA programming, get the Full Version (not just the runtime) version 5.0 or higher for the best support of the latest .NET 4.0 Framework, USB and TCP/IP. The driver is available from http://www.ni.com/visa/ or from NI Device Driver CD that comes with NI hardware and is installed along with LabWindows/CVI. Most LabWindows/CVI drivers use VISA to communicate.

VISA uses connection strings to set up communication with the VNA over various protocols. Here are some connection string examples:

```
//VXI-11 Connection string
"TCPIP0::192.168.1.7::INSTR";


//GPIB Connection string
"GPIB0::6::INSTR";


//USB Connection string (vendor::product::serial_number)
"USB0::0x0B5B::0xFFD0::MS4647B-12345::INSTR";
```

The beauty of using VISA is that the only thing that needs to be changed for any of these possible communication protocols is the connection string. The rest of the code should be exactly the same (except for SOCKETS which are not covered in this document). For TCP/IP we recommend using VXI-11 since it better implements the IEEE 488.2 standard and all status checking. The *Getting Started* section of the Programming Examples shows how to set up for communication over VXI-11 (TCP/IP).

Programming Environments

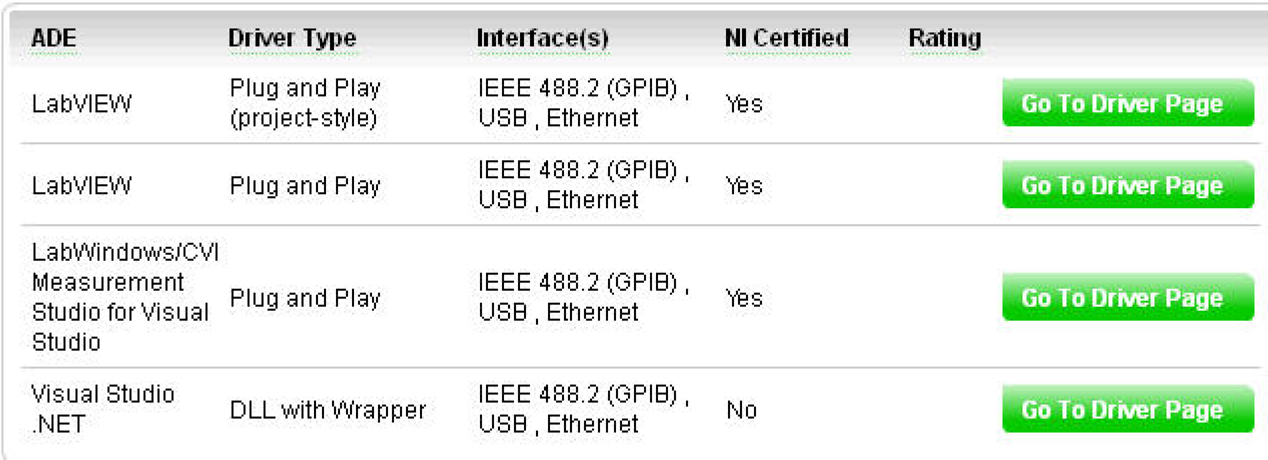
Programming environments that are commonly used with test equipment include LabVIEW, LabWindows/CVI, Microsoft Visual Studio, Visual Basic 6, HP Basic, and so on. NI-VISA supports the following Windows development environments:

- LabVIEW
- LabWindows/CVI
- Measurement Studio for Visual Studio (all versions)
- Microsoft Visual Studio 2010/2008/2005/2003/6.0
- Microsoft Visual Basic 6.0

For the examples below, we'll use LabWindows/CVI Version 8.5 with the Anritsu VectorStar au464x LabWindows/CVI driver. We'll interchangeably use the terms *LabWindows/CVI* and *CVI*. These both refer to the programming environment.

B-3 Installing the Anritsu au464x LabWindows/CVI Driver

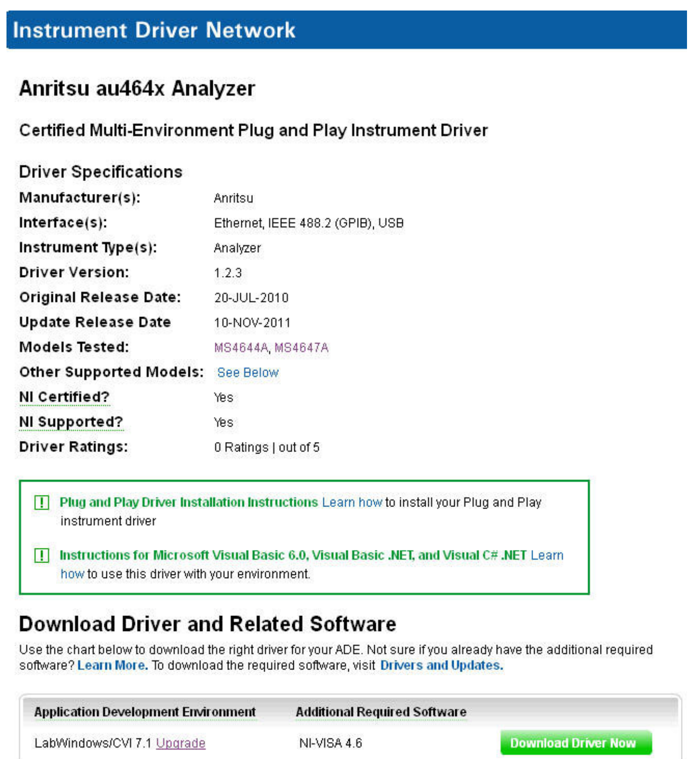
The Anritsu au464x CVI driver was developed and is supported by National Instruments. The driver is available at the National Instrument's Instrument Driver Network (http://www.ni.com/devzone/idnet/) – search for "MS464*" where the * is a wildcard character.



ADE	Driver Type	Interface(s)	NI Certified	Rating	
LabVIEW	Plug and Play (project-style)	IEEE 488.2 (GPIB) , USB , Ethernet	Yes		Go To Driver Page
LabVIEW	Plug and Play	IEEE 488.2 (GPIB) , USB , Ethernet	Yes		Go To Driver Page
LabWindows/CVI Measurement Studio for Visual Studio	Plug and Play	IEEE 488.2 (GPIB) , USB , Ethernet	Yes		Go To Driver Page
Visual Studio .NET	DLL with Wrapper	IEEE 488.2 (GPIB) , USB , Ethernet	No		Go To Driver Page

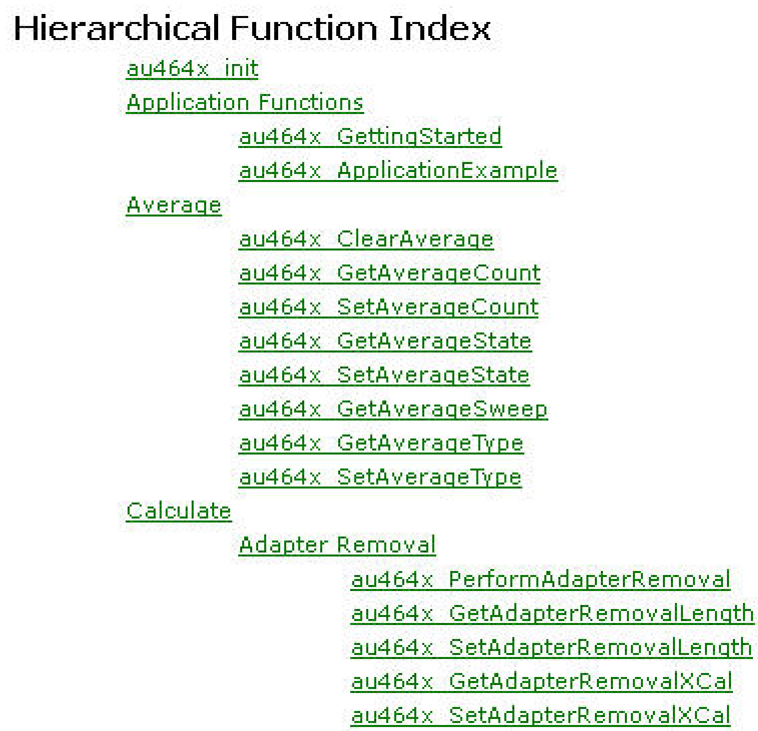
Here are the VectorStar VNA drivers available from the National Instrument's web site. We'll use the LabWindows/CVI Plug and Play Driver.

Figure B-1. VectorStar VNA Drivers Available at National Instruments



More details on the au464x driver for LabWindows/CVI. When you download the driver you'll get a zip file.

Figure B-2. au464x Driver Detailed Specifications



Hierarchical Function Index

- au464x_init
- Application Functions
  - au464x_GettingStarted
  - au464x_ApplicationExample
- Average
  - au464x_ClearAverage
  - au464x_GetAverageCount
  - au464x_SetAverageCount
  - au464x_GetAverageState
  - au464x_SetAverageState
  - au464x_GetAverageSweep
  - au464x_GetAverageType
  - au464x_SetAverageType
- Calculate
  - Adapter Removal
    - au464x_PerformAdapterRemoval
    - au464x_GetAdapterRemovalLength
    - au464x_SetAdapterRemovalLength
    - au464x_GetAdapterRemovalXCal
    - au464x_SetAdapterRemovalXCal

The driver includes an HTML help file

Figure B-3. au464x Driver HTML Help Files

au464x_init

```
ViStatus au464x_init (ViRsrc resourceName, ViBoolean iDQuery, ViBoolean resetDevice, ViPSession instrumentHandle);
```

Purpose

This function performs the following initialization actions:
- Opens a session to the Default Resource Manager resource and a session to the specified device using the interface and address specified in the Resource_Name control.
- Performs an identification query on the Instrument.
- Resets the instrument to a known state.
- Sends initialization commands to the instrument that set any necessary programmatic variables such as Headers Off, Short Command form, and Data Transfer Binary to the state necessary for the operation of the instrument driver.
- Returns an Instrument Handle which is used to differentiate between different sessions of this instrument driver.
- Each time this function is invoked a Unique Session is opened. It is possible to have more than one session open for the same resource.

Notes:
(1) If this instrument does not support an ID Query, and the ID Query control is set to "Do Query" then this function should return the Warning Code 0x3FFC0101 - VI_WARN_NSUP_ID_QUERY.
(2) If this instrument does not support a Reset, and the Reset control is set to "Reset Device" then this function should return the Warning Code 0x3FFC0102 - VI_WARN_NSUP_RESET.

Parameters

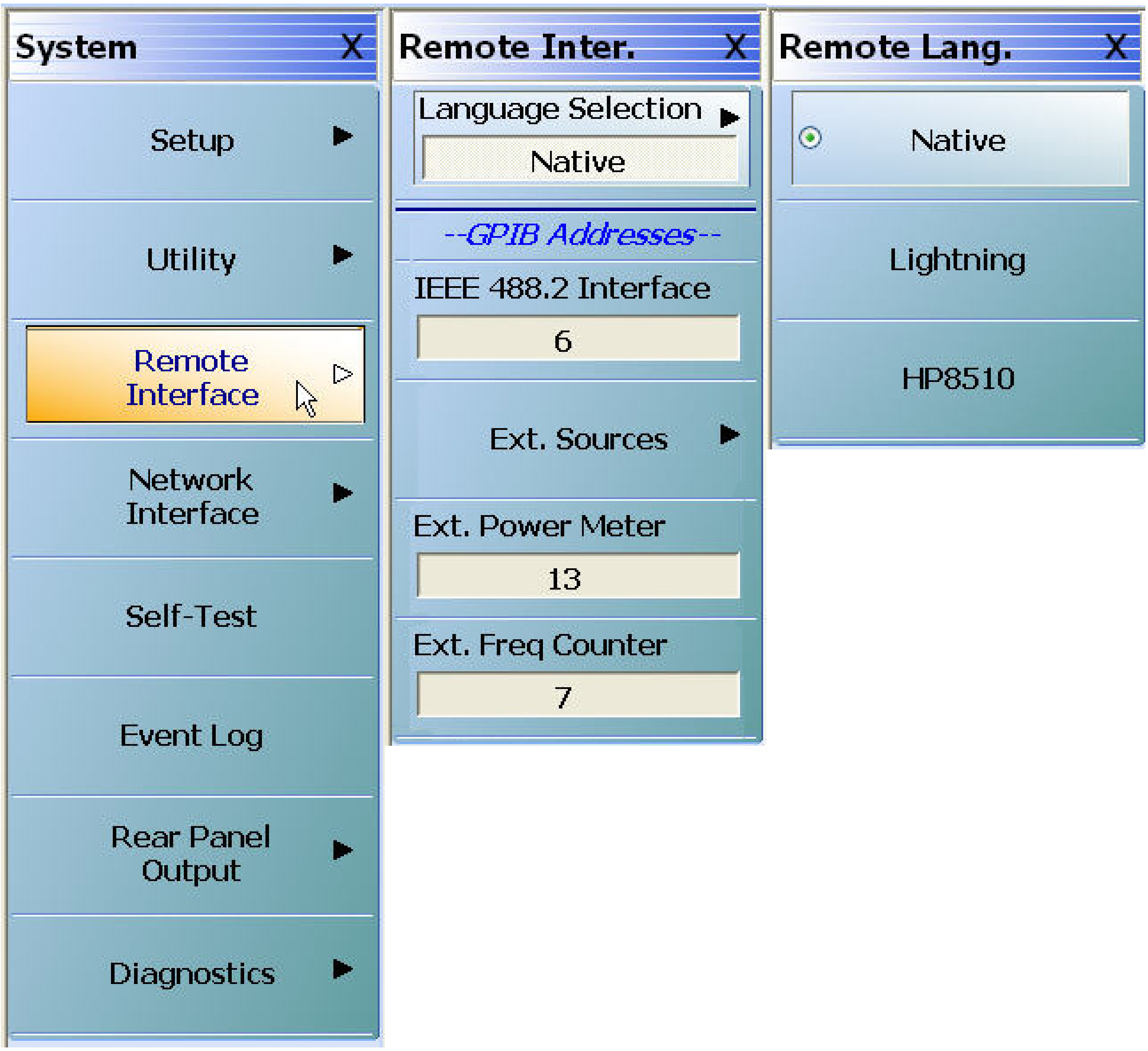
Input

Name	Type	Description
resourceName	ViRsrc	This control specifies the interface and address of the device that is to be initialized (Instrument Descriptor). The exact grammar to be used in this control is shown in the note below. Default Value: "GPIB::1" Notes: (1) Based on the Instrument Descriptor, this operation establishes a communication session with a device. The grammar for the Instrument Descriptor is shown below. Optional parameters are shown in square brackets ([]). Interface Grammar -- GPIB GPIB[board]::primary address[::secondary address] [::INSTR] The GPIB keyword is used with GPIB instruments. The default value for optional parameters are shown below. Optional Parameter Default Value --- board 0 secondary address none - 31

The help for each function is pretty detailed as this example shows.

Figure B-4. au464x Driver HTML Help File Detail

Setup VectorStar for Native Language

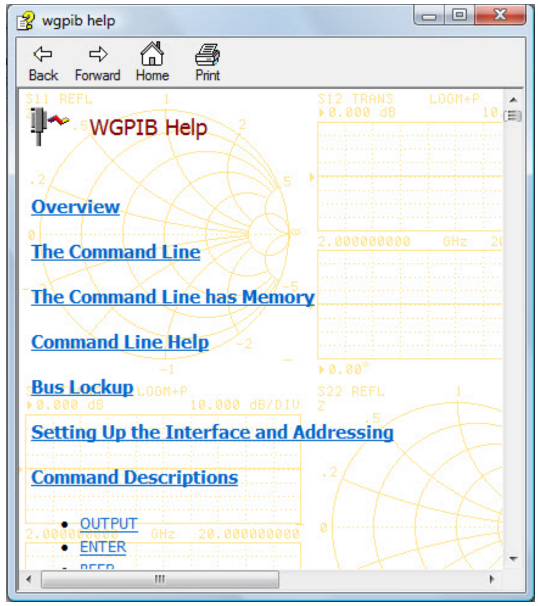


Use these menus to set up VectorStar to use Native SCPI commands for GPIB communication. We can also set this in our CVI program and we'll do that in Examples 4, 7, and 8.

Figure B-5. VectorStar REMOTE LANGUAGE Setup Menus

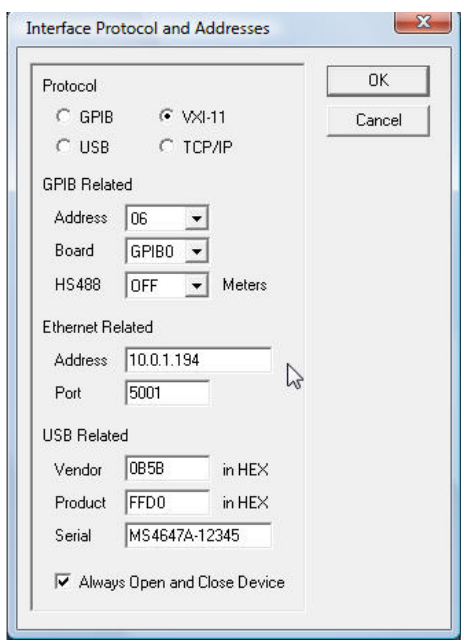
B-4 Quick Start with the Windows GPIB Utility

Windows GPIB (WGPIB) is a useful windows program that is available from the Anritsu web site. The latest version is 3.75 and you'll want to get that one or higher to get the latest communication functions. Before writing any software, make sure you can write to and read from VectorStar over the communication protocol you're interested in.



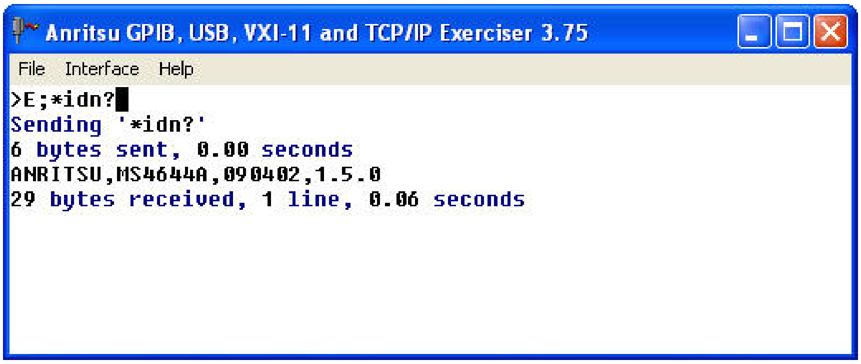
Take a quick look at WGPIB Help to get an overview of the application.

Figure B-6. Windows GPIB Utility Help



Example of setting up a VXI-11 connection to VectorStar. The VectorStar TCP/IP address is 10.0.1.194 in this example.

Figure B-7. VXI-11 Connection to VectorStar with TCP/IP



The result of using WGPIB to query the instrument using the "*IDN?" command.

Figure B-8. Using WGPIB with *IDN? Command

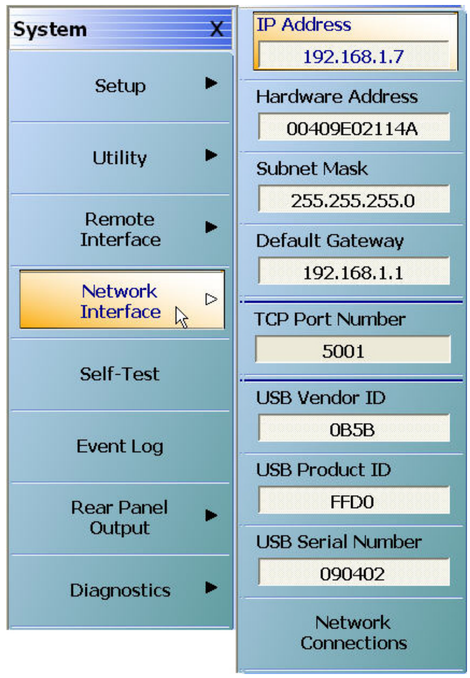
References

- Getting Started with LabWindows/CVI
- Lightning Programming Manual – 10410-00262
- VectorStar Programming Manual – 10410-00322
- VectorStar Programming Manual Supplement – 10410-00323

B-5 Programming Examples

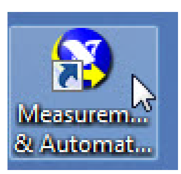
Getting Started

We'll create a few simple programs to demonstrate the use of the au464x CVI driver for controlling VectorStar. There are a few necessary steps to take before getting started with the programs.



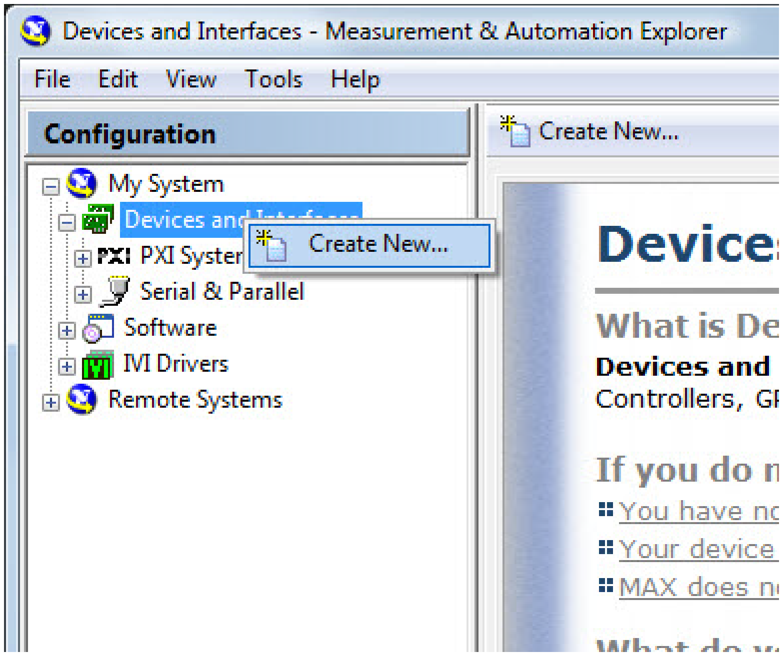
We want to communicate to VectorStar using VXI-11 (TCP/IP) so we'll need to note the IP Address of the VNA and set up a resource (a connection string).

Figure B-9. Setup – Obtaining VNA IP Address



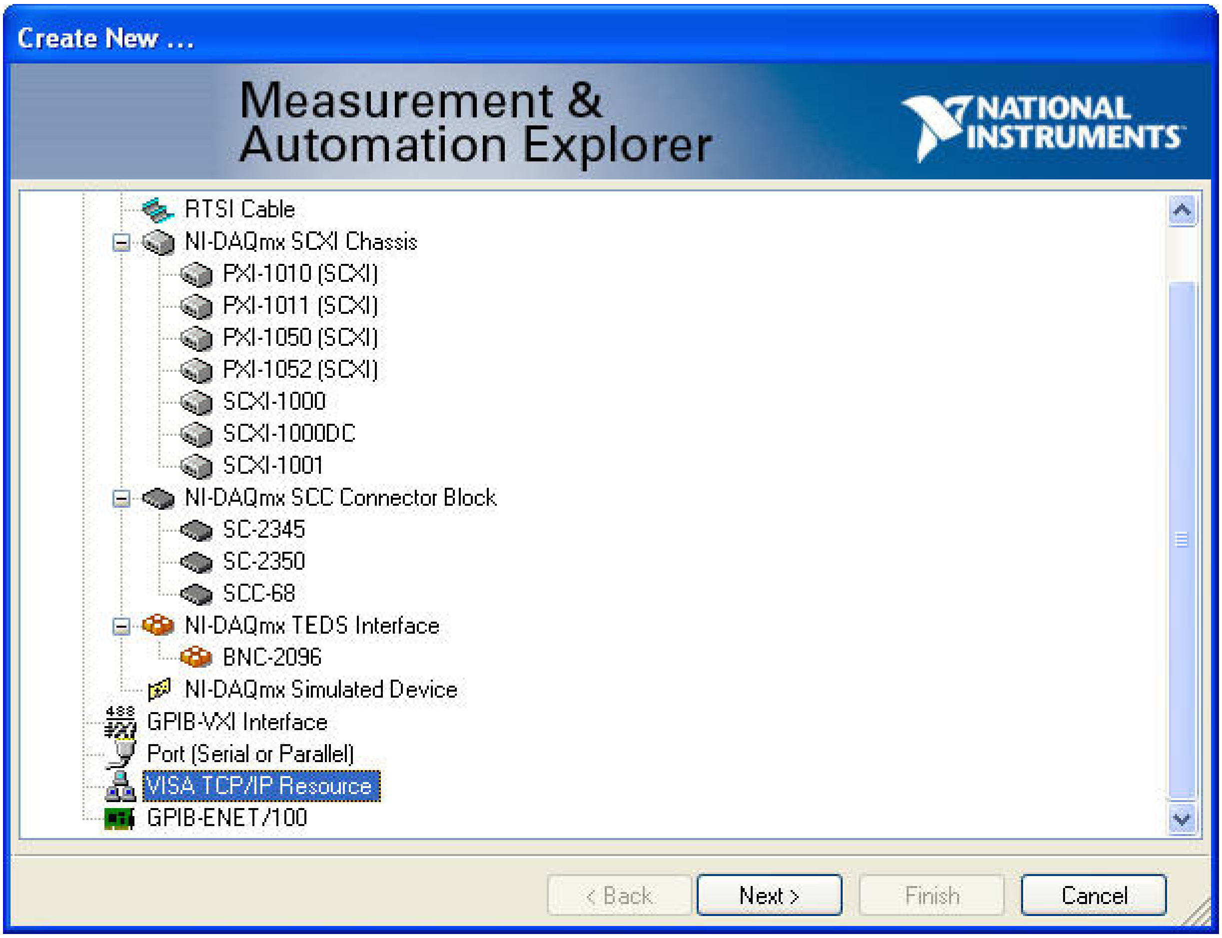
We'll set up a resource in NI-MAX. The programs will reference this resource rather than a specific address.

Figure B-10. Setup – Resource Setup in NI-MAX



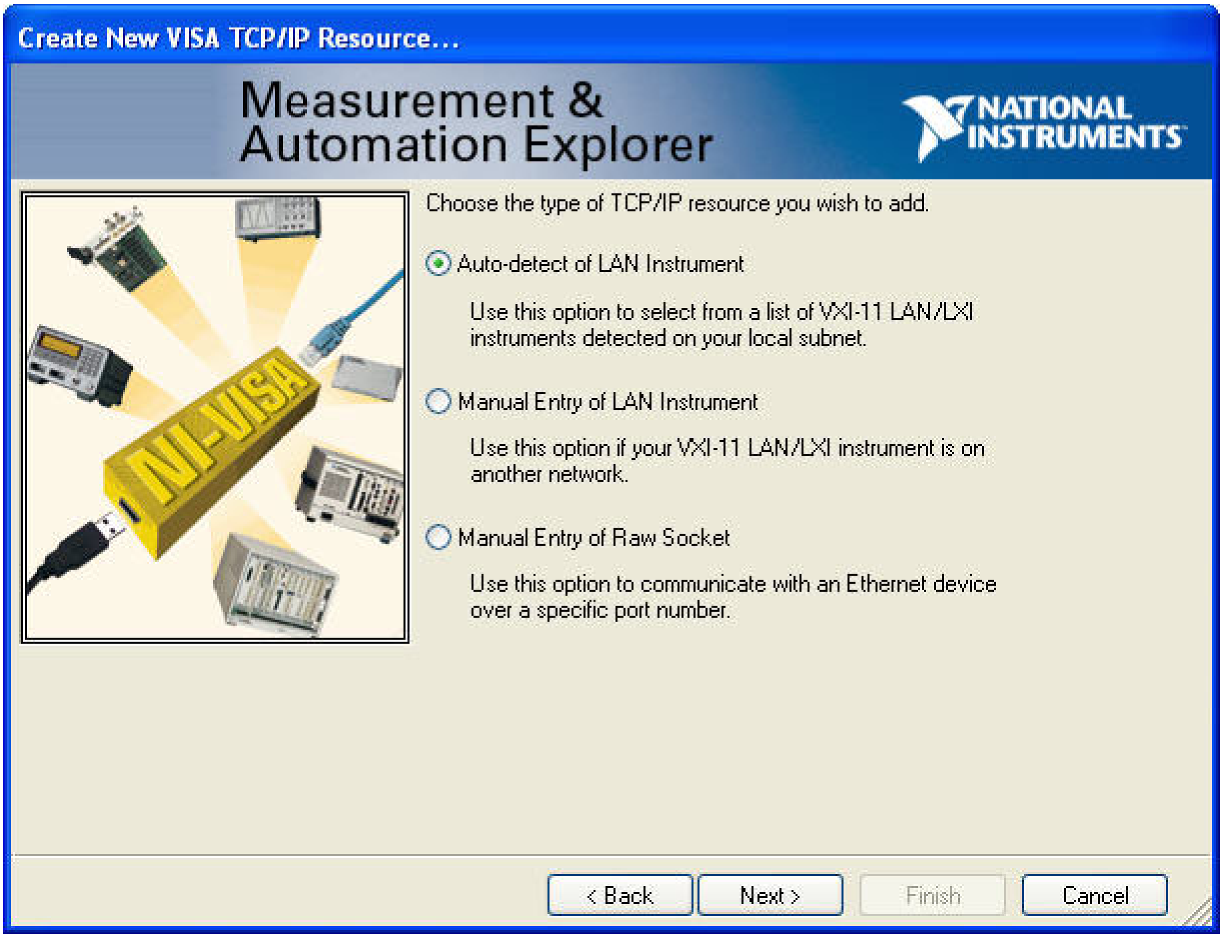
Create a new resource.

Figure B-11. Setup – Create New Resource



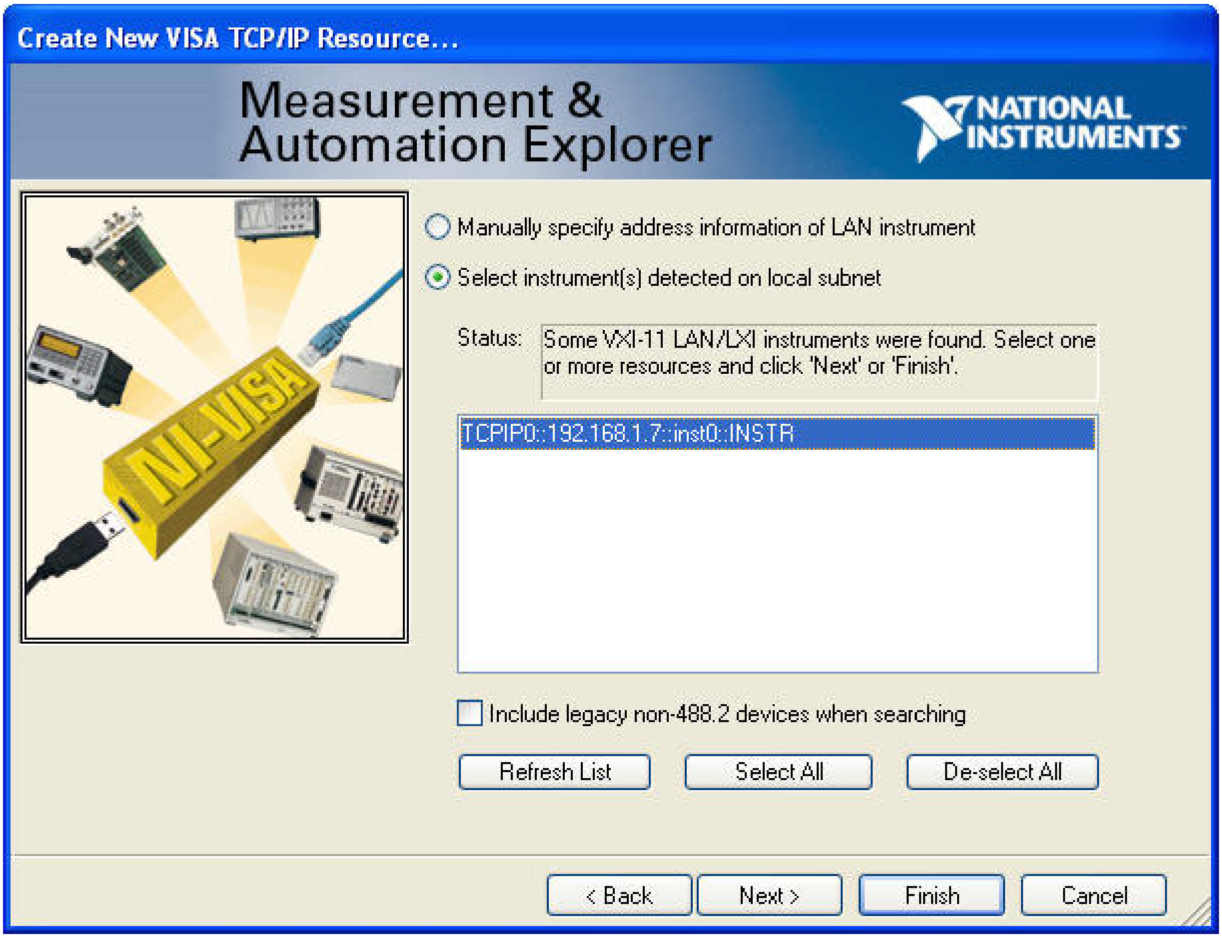
Select VISA TCP/IP Resource.

Figure B-12. Setup – VISA TCP/IP Resource



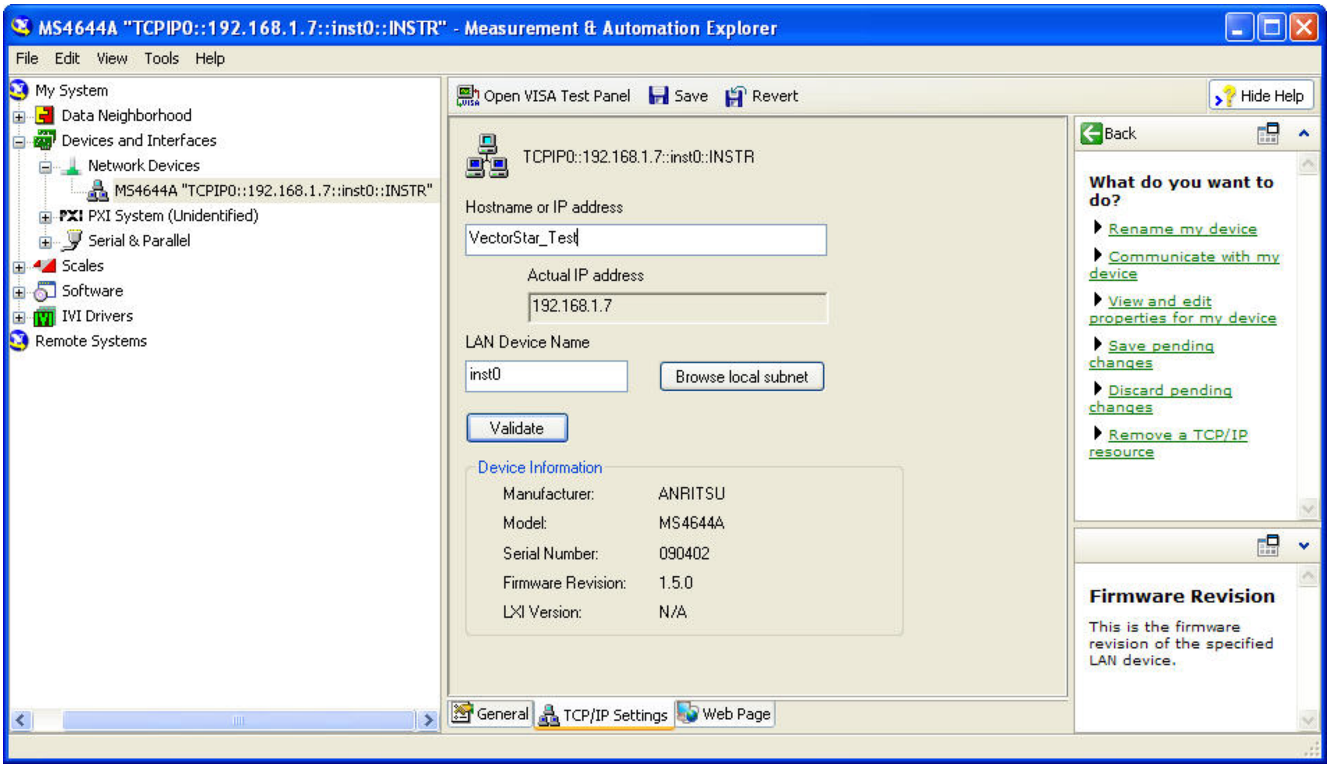
If the controlling pc and the VNA are on the same local sub-network (this is usually true if the first 3 numbers in the IP address are the same — for example, 192.168.1.x in this case), then you can probably Auto-detect the VNA.

Figure B-13. Setup – VISA TCP/IP – Auto-Detect Selected



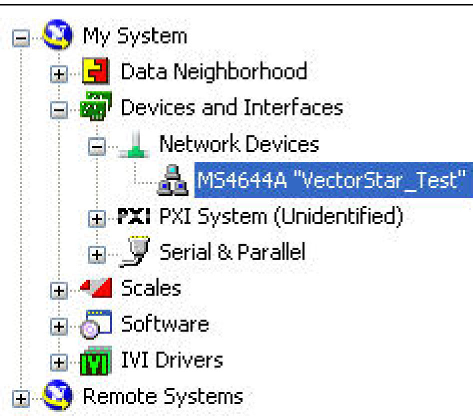
Select the detected instrument.

Figure B-14. Setup – VISA TCP/IP – Select detected instrument



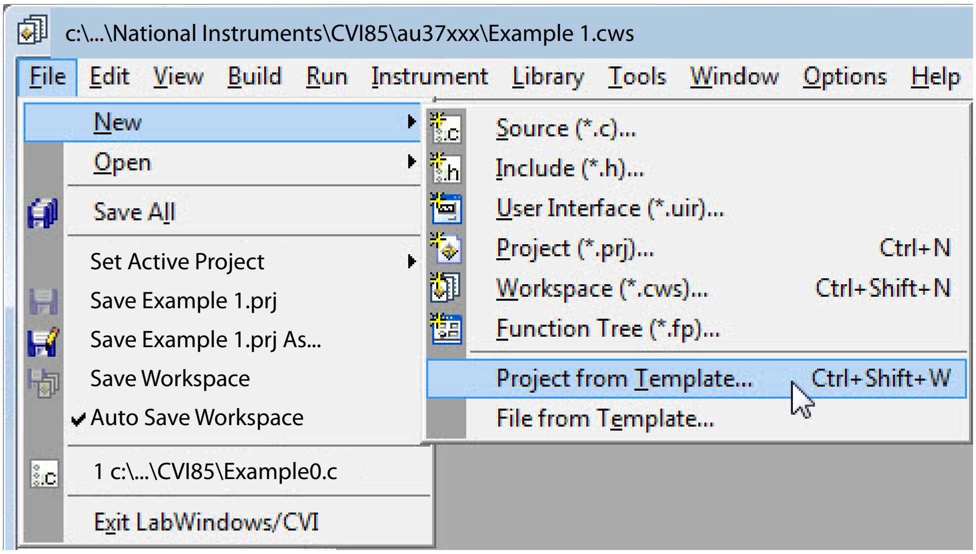
Give the instrument an alias. Later on we'll use this alias in LabWindows/CVI.

Figure B-15. Setup – VISA TCP/IP – Assign Alias



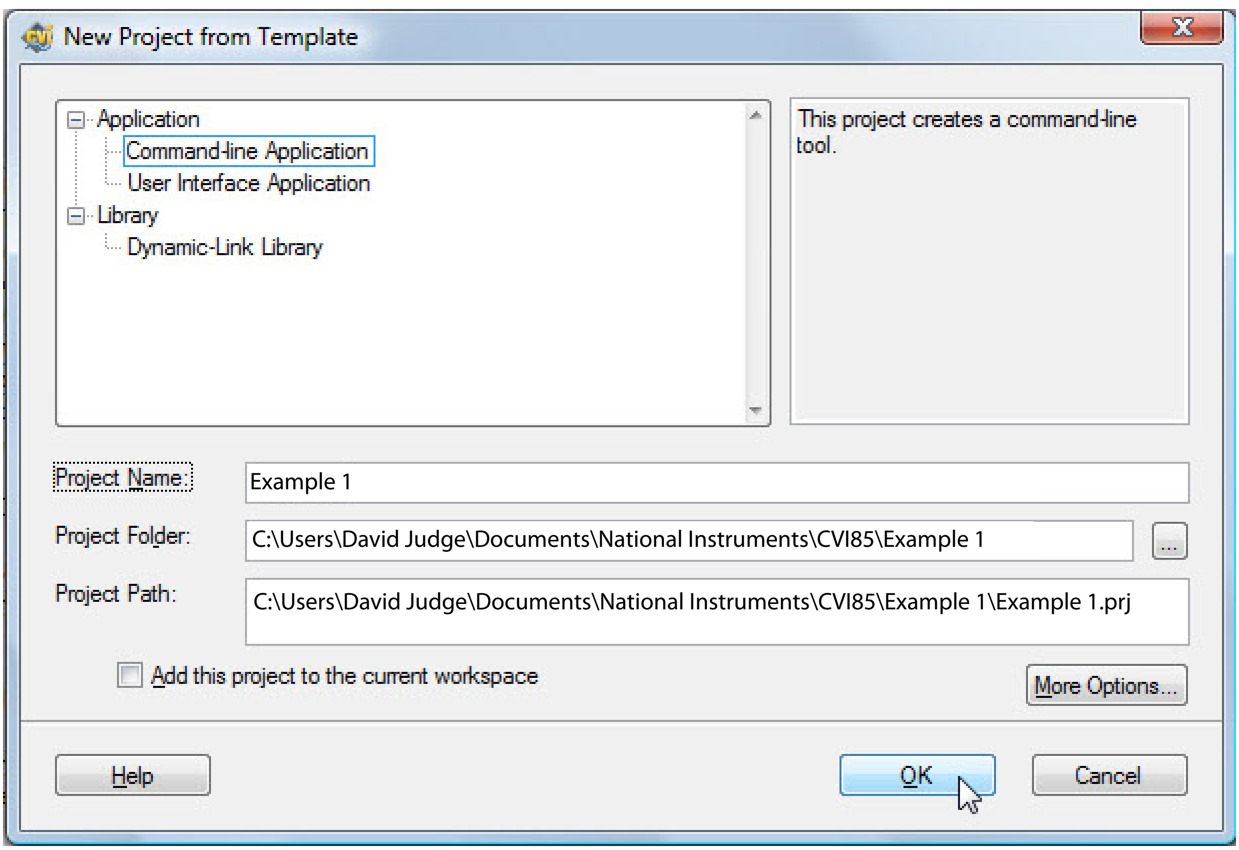
Note that the VISA connection string has been replaced with the VISA alias.

Figure B-16. Setup – VISA Alias in use



Now back in LabWindows/CVI we're ready to start creating the first example. We'll create a new Project from Template.

Figure B-17. Creating the Project

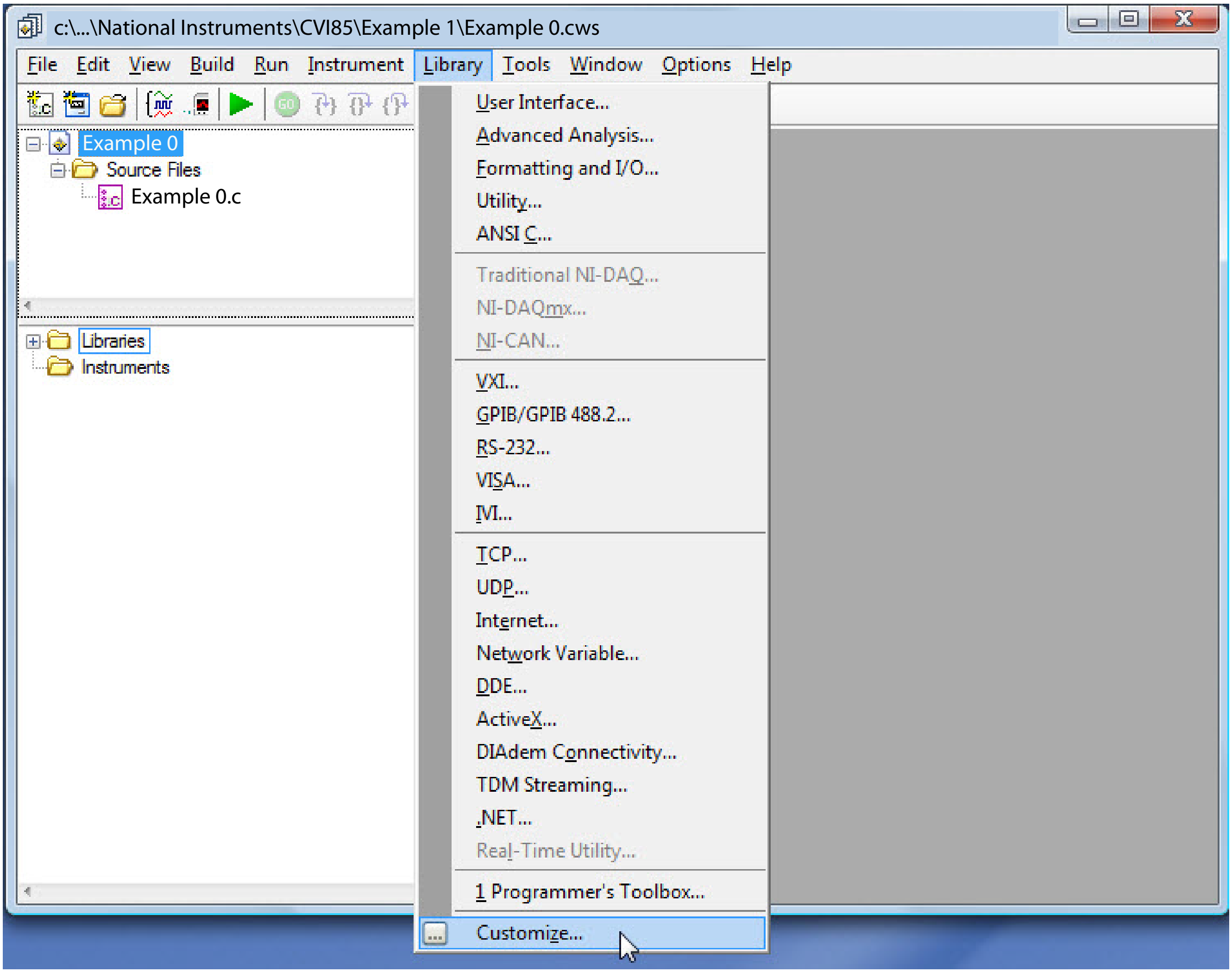


We'll use the Command-line Application Template.

Figure B-18. Command-Line Template

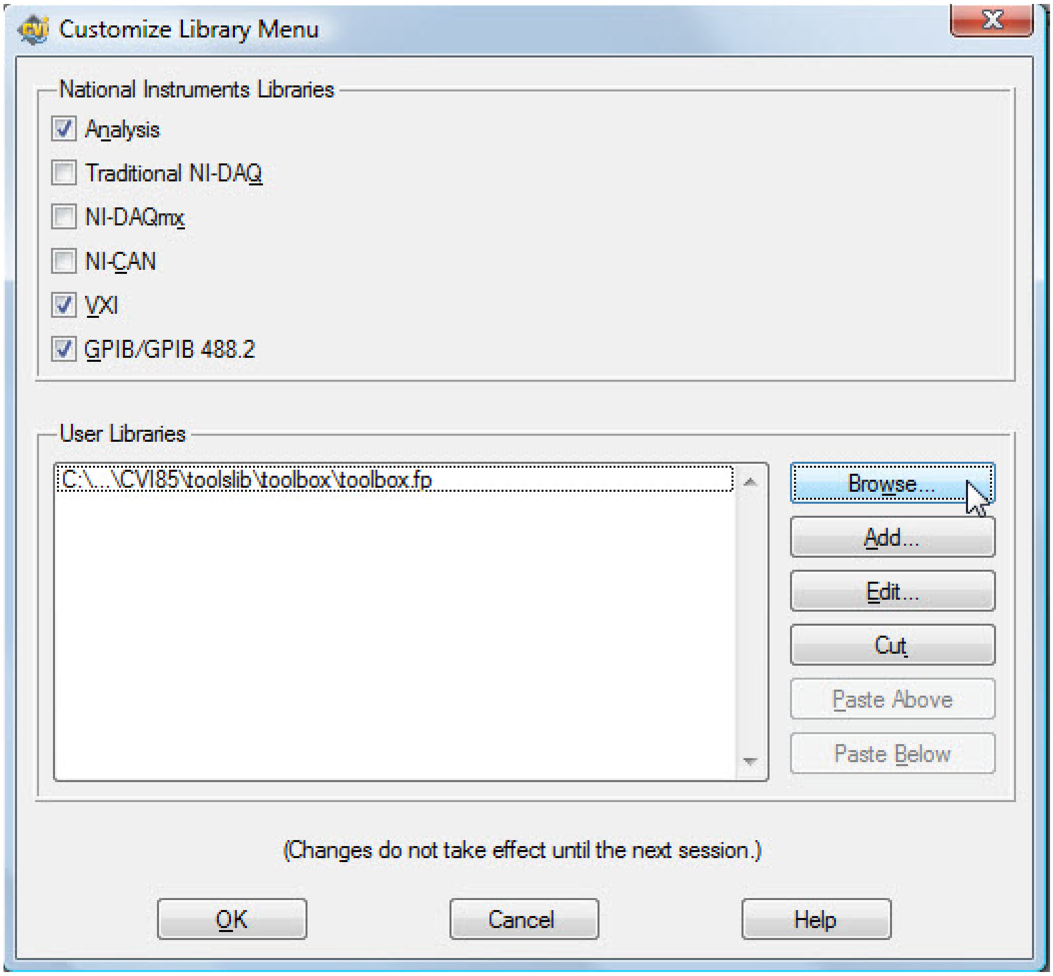
Setting up the Driver to Load every time CVI starts

Since we're planning to do a lot of VNA programs we'll set up CVI to load the Anritsu driver automatically every time CVI starts up.



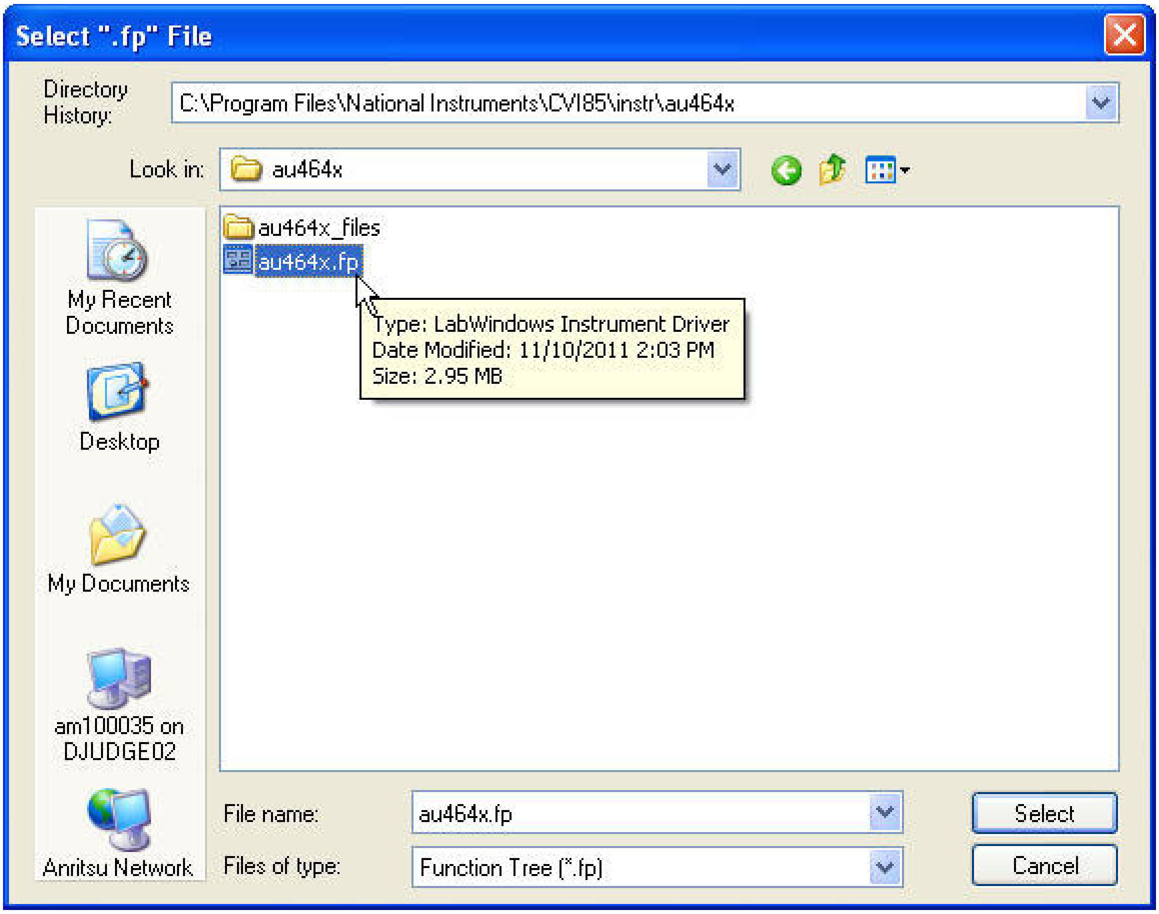
Go to Library | Customize to have CVI load the driver into our user Library every time CVI starts. This is a preferable way to have access to the driver.

Figure B-19. Auto-Load Anritsu Driver



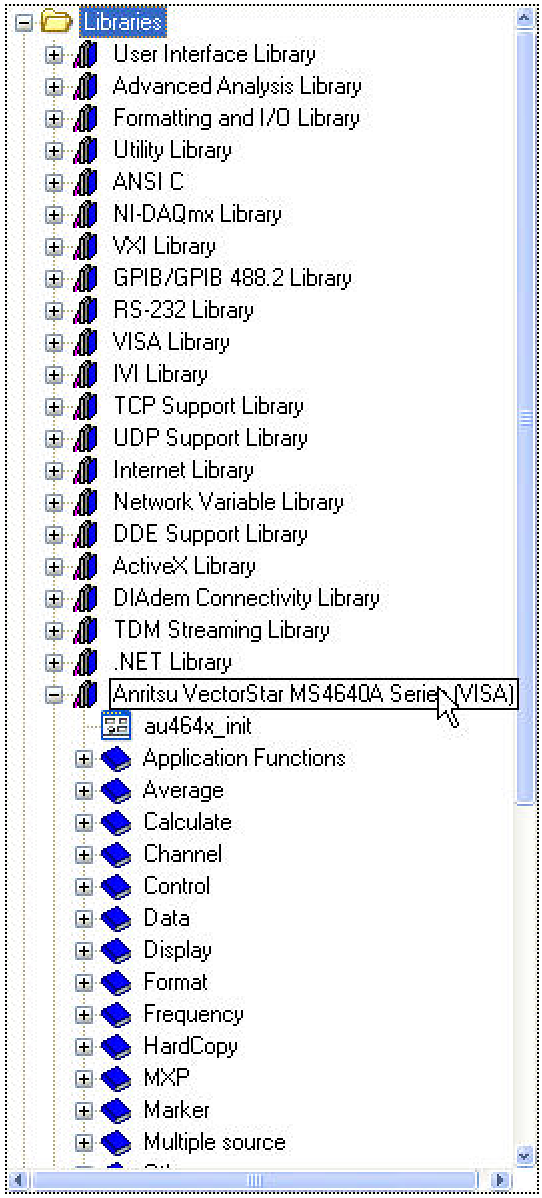
We'll Browse to find the au464x.fp file.

Figure B-20. Find the au464x.fp File



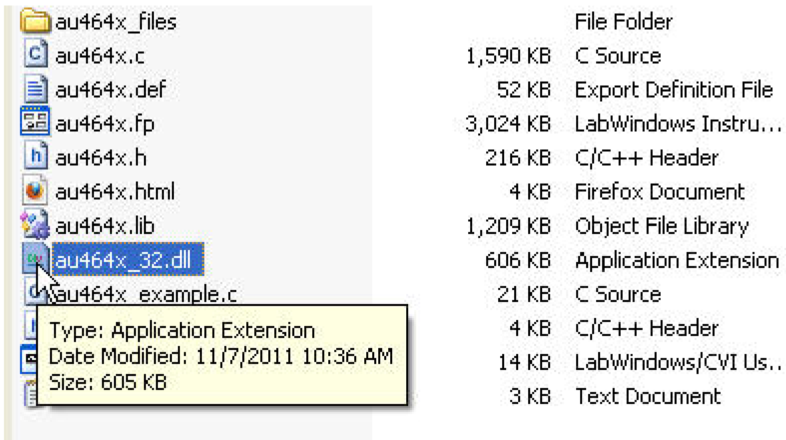
Add the Function Tree (".fp") file.

Figure B-21. au464x.fp Function Tree File



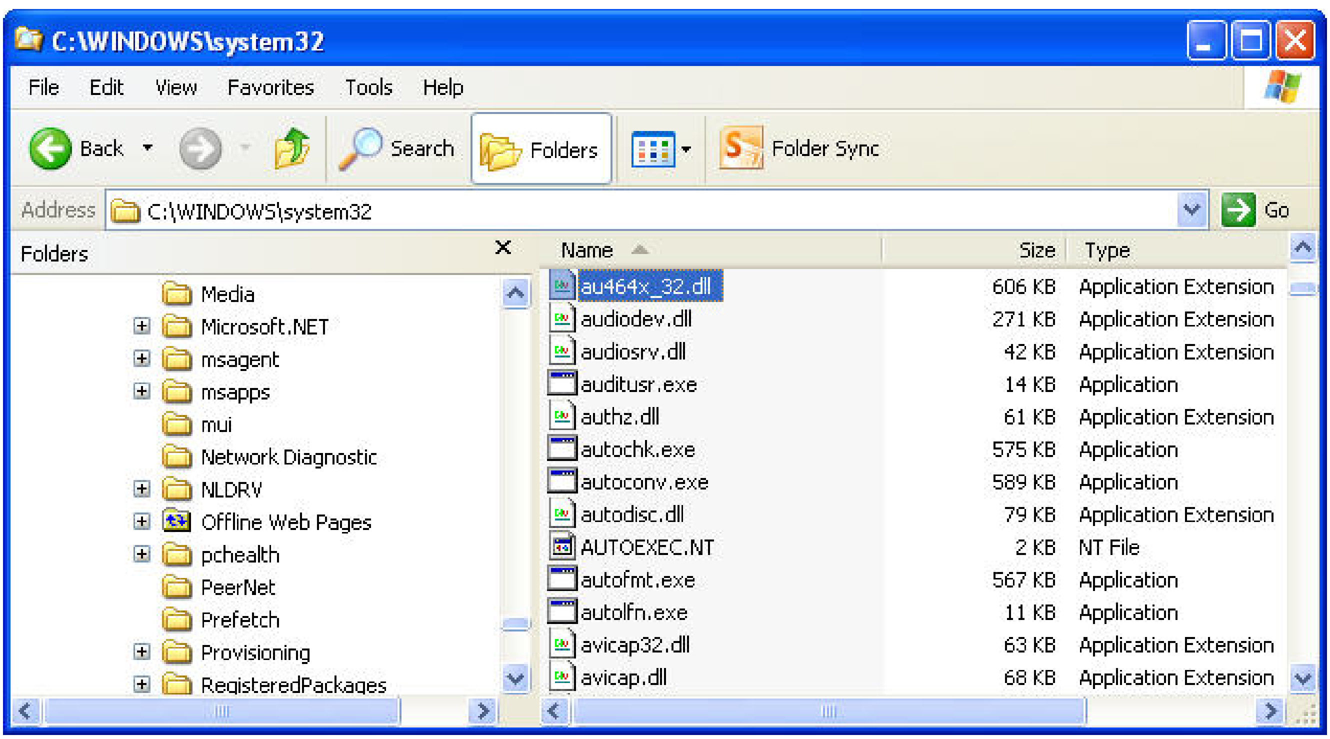
The driver is now in the Libraries folder.

Figure B-22. Libraries Folder



Finally, we'll need to copy the driver DLL to someplace on the on the system path. If you don't know or care what the system path is, then just copy the file to C:\Windows\System32.

Figure B-23. Find the au464x_32.dll DLL Driver



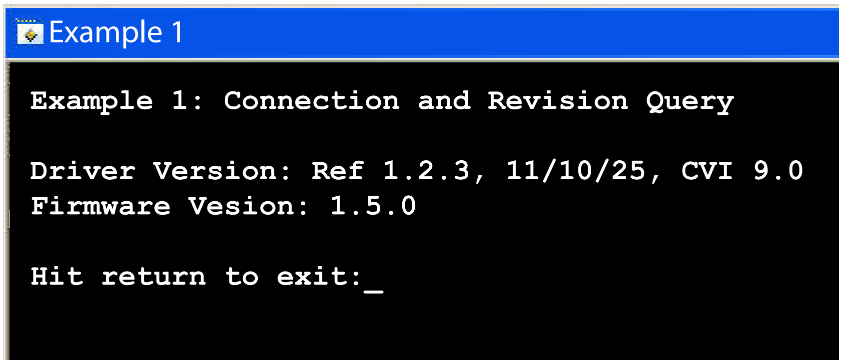
Here we're copying the DLL to C:\Windows\System32. This step seems strange but it's necessary. There are other places to put the DLL, if you're interested, so check the LabWindows/CVI documentation.

Figure B-24. Copy the au464x_32.dll DLL Driver

B-6 Example 1 – Open a Session to Obtain Instrument Information

In this first example we'll simply open a communication session to the VNA and then use two of the driver VIs to get some information about the VNA. The code snippet is below.

```
//=============================================================================
//
// Title:       Example 1
// Purpose:     Open session to obtain instrument information
//
// Created on:  11/30/2011 by David Judge.
// Copyright:   Anritsu. All Rights Reserved.
//
//=============================================================================

//=============================================================================
// Include files
#include <ansi_c.h>
#include "au464x.h"


int main (int argc, char *argv[])
{
    ViSession session;
    ViStatus status;
    ViChar d[256];
    ViChar d1[256];

    printf("Example 0: Connection and Revision Query\n\n");
    //status = au464x _init ("TCPIP0::192.168.1.7::INSTR", VI_TRUE, VI_FALSE, 
&session);
    status = au464x_init ("VectorStar_Test", VI_TRUE, VI_FALSE, &session);
// We could use a VISA connection in string in the au464x_init function above
// but we prefer to use the VISA alias (“VectorStar_Test”) we set up earlier.
    status = au464x_revisionQuery (session, d,d1);
    au464x_close(session);
    printf("Driver Version: %s\n",d);
    printf("Firmware Version: %s\n",d1);

    printf("\n\nHit return to exit:");
    getc(stdin);

    return 0;
}
```



The results of running Example 1. The au464x_revision_query() function returns two strings. The first is the version of the driver and the second is the version of firmware on the VNA.

Figure B-25. Example 1 – Output Results

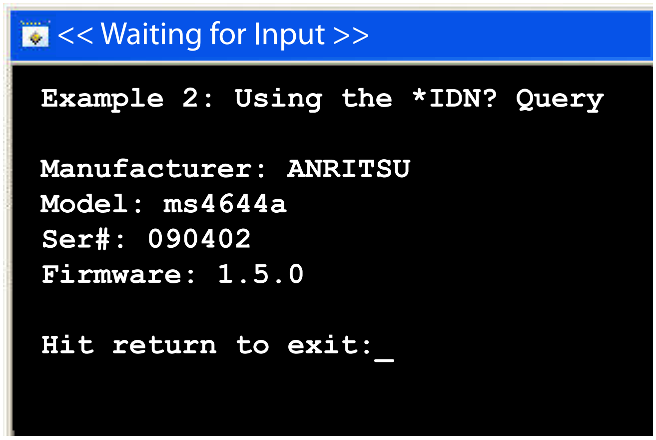
B-7 Example 2 – Sending the *IDN? Command and Displaying results

The previous example used only driver functions to get some information from the VNA. The GPIB command, "*IDN?" returns the Manufacturer, Model #, Serial Number and Firmware Version. We used this command previously in Figure B-8, "Using WGPIB with *IDN? Command" on page B-8 above using the WGPIB utility. In this example we directly issue the "*IDN?" command and then parse the different parts of the response string. The code snippet is below.

```
//================================================================================
//
// Title:       Example 2
// Purpose:     Sending the *IDN? command and displaying results
//
// Created on:  11/30/2011 by David Judge.
// Copyright:   Anritsu. All Rights Reserved.
//
//================================================================================

//================================================================================
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include "au464x.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au464x_status = checkErr((fCal)), au464x_status < VI_SUCCESS) \
        goto Error; else
// We’ll use this CHECKERR Macro above in all the examples from now on.
// It provides a convenient way to check every function for an error.

int main (int argc, char *argv[])
{
    ViStatus status;
    ViUInt32 read_count;
    ViStatus    au464x_status   = VI_SUCCESS;
    ViChar l_buffer[50];
    ViChar* p2Manf = NULL;
    ViChar* p2Model = NULL;
    ViChar* p2Ser = NULL;
    ViChar* p2Firm = NULL;

    printf("Example 1: Using the *IDN? Query\n\n");
    CHECKERR(au464x_init ("VectorStar_Test", VI_TRUE, VI_FALSE, &session));
    CHECKERR(au464x_writeInstrData (session, "*IDN?"));
```

```
    CHECKERR(viRead (session, (ViPBuf)l_buffer, 50, &read_count));
    l_buffer[49] = '\0';
    au464x_close(session);
    p2Manf = strtok(l_buffer, ",");
    p2Model = strtok(NULL, ",");
    p2Ser = strtok(NULL, ",");
    p2Firm = strtok(NULL, "\n");
// We use au464x_writeInstrData() above to directly send a GPIB command.
// Then we use viRead() to read the results.

    printf("Manufacturer: %s\nModel: %s\nSer#: %s\nFirmware: %s\n",
        p2Manf,p2Model,p2Ser,p2Firm);



    printf("\n\nHit return to exit:");
    getc(stdin);

    return 0;

Error:
    printf("\n\nDetected an Error--Hit return to exit:");
    getc(stdin);

    return 0;
}



ViStatus checkErr (ViStatus status)
// The checkErr() function is called from the MACRO code.
// We check the status of the driver function and report any errors.

{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];

    if (status < VI_SUCCESS)
    {
        au464x_errorMessage (session, status, error_message);
        sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status, error_message);
        MessagePopup ("Error", error_buffer);
```

```
        au464x_errorQuery (session, error_code, error_message);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        MessagePopup ("Error", error_buffer);
        au464x_close(session);
        session = 0;
    }
    return status;
}
```



Output from Example 2. We use the au464x_ writeInstrData() function to directly send GPIB commands. We then use the VISA function viRead() to read the results.

Figure B-26. Example 2 – au464x_writeInstrData() and *IDN? – Output

B-8 Example 3 – Error Checking

This example shows that if an invalid GPIB string is sent to the VNA then the CHECKERR macro catches the error and displays the error message from the VNA. Here we send two valid strings: "*IDN?" and then "OID". But the third string is not a valid GPIB command and the instrument reports this. The code snippet is below.

```
//================================================================================
//
// Title:       Example 3
// Purpose:     Error Checking
//
// Created on:  11/30/2011 by David Judge.
// Copyright:   Anritsu. All Rights Reserved.
//
//================================================================================

//================================================================================
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include "au464x.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au464x_status = checkErr((fCal)), au464x_status < VI_SUCCESS) \
        goto Error; else


int main (int argc, char *argv[])
{

    ViStatus status;
    ViUInt32 read_count;
    ViStatus    au464x_status   = VI_SUCCESS;
    ViChar l_buffer[50];
    ViChar* p2Manf = NULL;
    ViChar* p2Model = NULL;
    ViChar* p2Ser = NULL;
    ViChar* p2Firm = NULL;

    printf("Example 2: Testing for Errors\n\n");
    CHECKERR(au464x_init ("VectorStar_Test", VI_TRUE, VI_FALSE, &session));
```

```
    //Use Native VNA Error Checking
    CHECKERR(au464x_writeInstrData (session, "LANG NATIVE"));

    //First send a known good command
    CHECKERR(au464x_writeInstrData (session, "*IDN?"));
    CHECKERR(viRead (session, (ViPBuf)l_buffer, 50, &read_count));

    //Send an OID - also a good command
    CHECKERR(au464x_writeInstrData (session, "OID"));
    CHECKERR(viRead (session, (ViPBuf)l_buffer, 50, &read_count));

    //This command is not a valid VNA command and should generate an error.
    CHECKERR(au464x_writeInstrData (session, "ABC"));
    au464x_close(session);

    printf("\n\nHit return to exit:");
    getc(stdin);

    return 0;

Error:
    printf("\n\nDetected an Error--Hit return to exit:");
    getc(stdin);

return 0;
}



ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViUInt32 read_count;
    ViChar  error_buffer [1024];


    ViInt32 my_error_code = 0;
    ViInt32* error_code = &my_error_code;
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4; //This means there is an error
```

```
    ViUInt16 VNA_ERROR_MSG = 16; //This means there is a message in the error buffer


    if (status >= 0)
        viReadSTB (session, &stb);


    // Check if stb & VNA_ERROR is set
    // Add one more check in this function to look at the status byte for
    // something in the error queue (bit 2) and also Message Available (bit 4).
    if (status < VI_SUCCESS | |  (((stb & VNA_ERROR) > 0) && ((stb & VNA_ERROR_MSG) >
0)))
    {
        au464x_writeInstrData (session, ":SYST:ERR?");
        viRead (session, (ViPBuf)error_message, 256, &read_count);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
        au464x_writeInstrData (session, "*CLS");
    }
    return status;
}
```

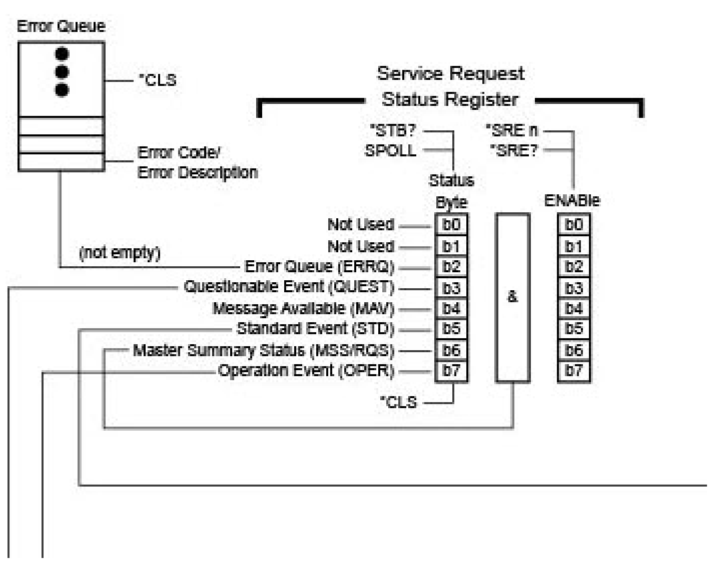
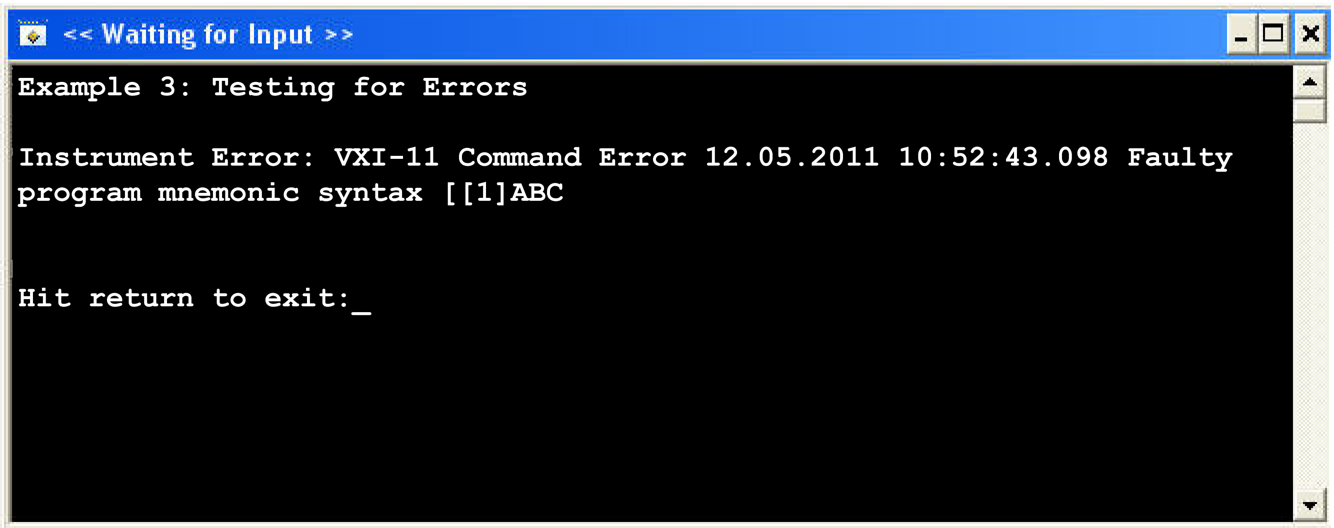


Figure B-27. Example 3 – Service Request Status Register – VectorStar Status Register

The Service Request Status Register. It is slightly changed from Lightning to VectorStar. If LANG LIGHT is set then the Lightning configuration of the Status Register is used. In this example we used “LANG NATIVE” so the VectorStar status register is used. The code checks b2 to see if the error queue is not empty and also b4 to see if a message is available.

Figure B-27. Example 3 – Service Request Status Register – VectorStar Status Register



We catch the command error after sending the bogus “ABC” command and report the message (Faulty program mnemonic syntax).

Figure B-28. Example 3 – Error Checking

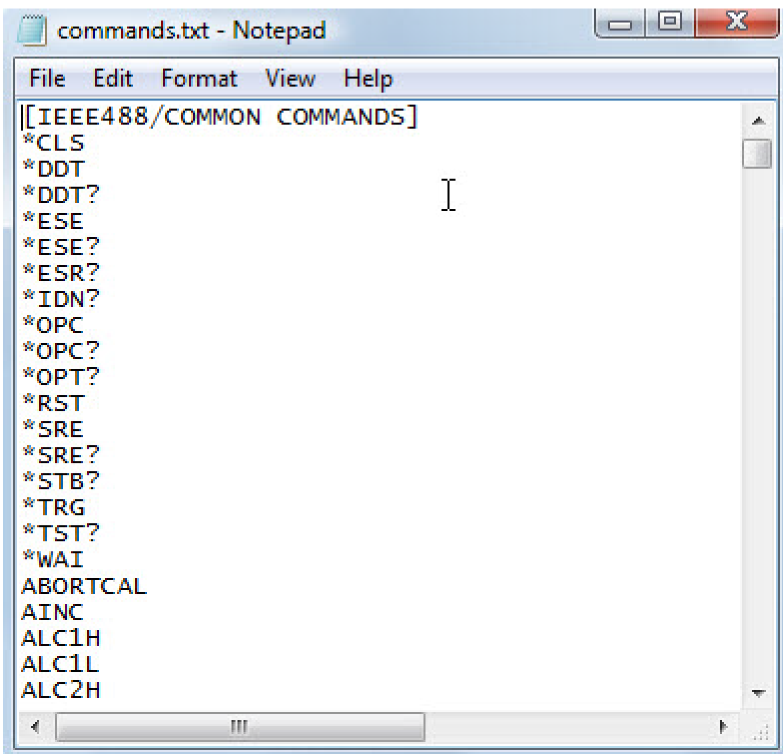
B-9 Example 4 — LIST Command – Send to a File

Here we send the Lightning commands “FMT1;LIST” to get the full list of commands supported by VectorStar. The sting returned is an ASCII arbitrary block so we use the au464x_readArbitraryBlock() function to strip off the arbitrary block header. Results are sent to a file. The code snippet is below.

```
//==============================================================================
//
// Title:       Example 4
// Purpose:     LIST
//
// Created on:  11/30/2011 by David Judge.
// Copyright:   Anritsu. All Rights Reserved.
//
//==============================================================================

//==============================================================================
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au464x.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au464x_status = checkErr((fCal)), au464x_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{

    ViInt32 retCount;

    ViStatus status;
    ViUInt32 read_count;
    ViStatus au464x_status = VI_SUCCESS;
    int fileHandle;
    static ViChar readBuffer[100000];
    // For these large buffers, we define them as static so
    // the memory is on the heap and not the stack.
    // Otherwise the stack may overflow
    // (unless you increase the stack size using Options | Build Options).
```

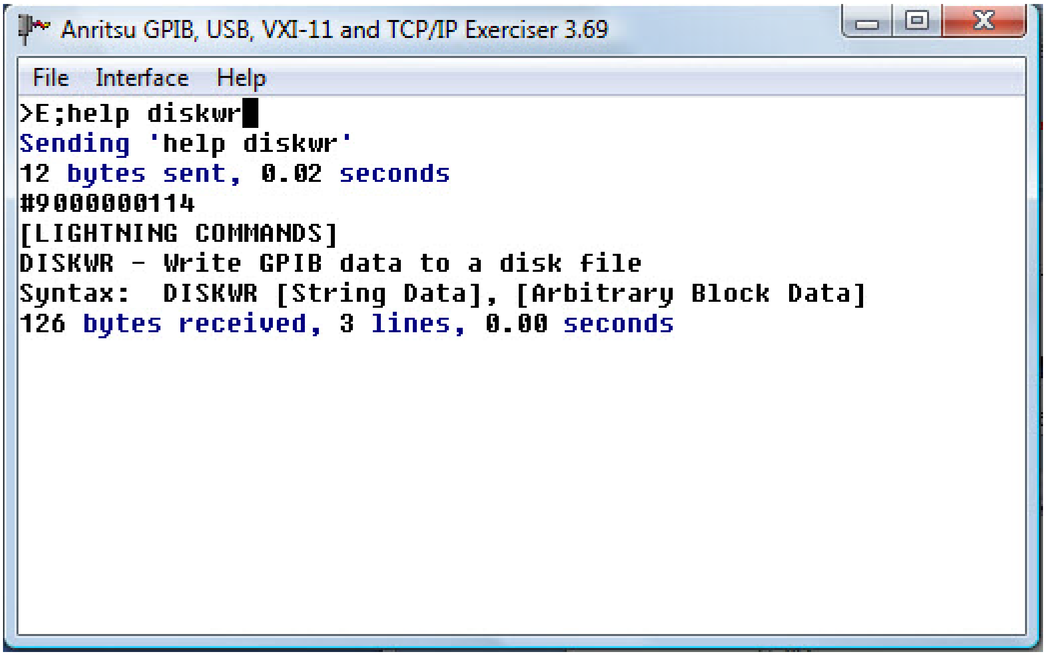
```
    CHECKERR(au464x_init ("VectorStar_Test", VI_TRUE, VI_FALSE, &session));
    CHECKERR(au464x_writeInstrData(session,"FMT1;LIST"));

    au464x_readArbitraryBlock(session, readBuffer, 1, 100000, VI_TRUE, &retCount);
    au464x_close(session);


    fileHandle = OpenFile (".\\commands.txt", VAL_WRITE_ONLY, VAL_OPEN_AS_IS,
VAL_ASCII);
    WriteFile (fileHandle, readBuffer, retCount);
    CloseFile (fileHandle);


Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}


ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViUInt32 read_count;
    ViChar  error_buffer [1024];


    ViInt32 my_error_code = 0;
    ViInt32* error_code = &my_error_code;
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4; //This means there is an error
    ViUInt16 VNA_ERROR_MSG = 16; //This means there is a message in the error buffer
(4+16)


    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set
    if (status < VI_SUCCESS ||  (((stb & VNA_ERROR) > 0) && ((stb & VNA_ERROR_MSG) >
0)))
```

```
    {
        au464x_writeInstrData (session, ":SYST:ERR?");
        viRead (session, (ViPBuf)error_message, 256, &read_count);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
    au464x_writeInstrData (session, "*CLS");

    }
    return status;
}
```



The List of all commands supported by VectorStar.

Figure B-29. VectorStar Supported Command List



Use WGPIB to get more help on any command. Help will tell you what type of command (Lightning, Native, 8510) you're asking about and provides syntax.

Figure B-30. WGPIB Help

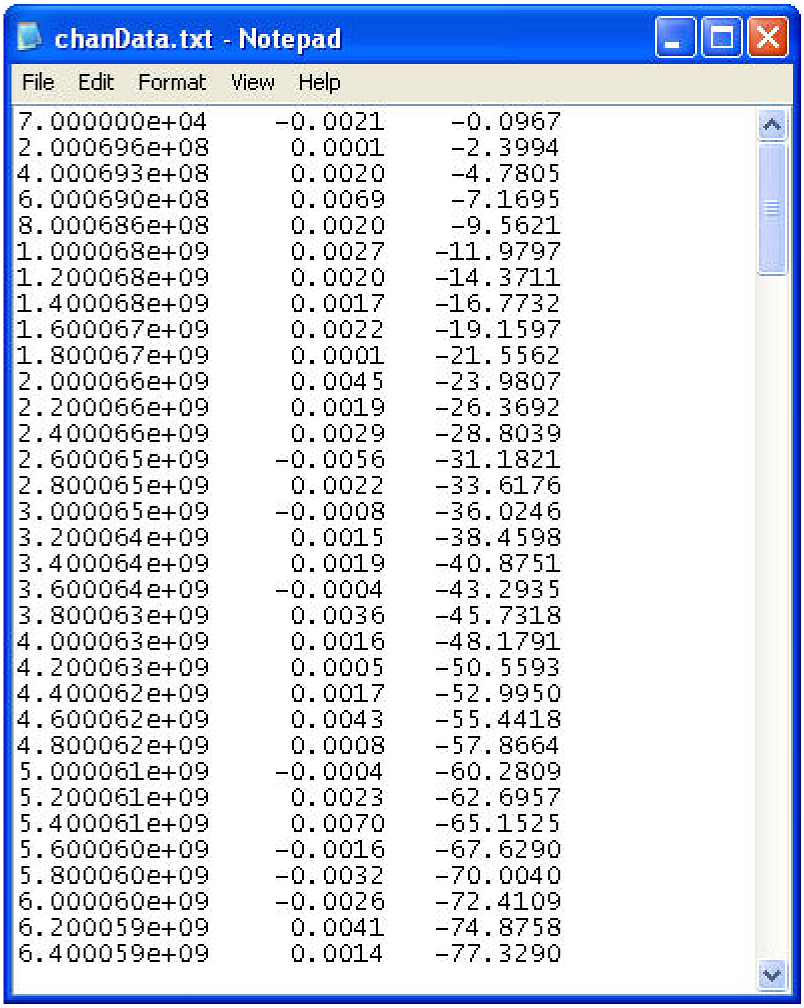
B-10 Example 5 — Acquiring Trace Data

Here we get the final data from Trace 2 which is set to Log magnitude and Phase data. The data comes out in a one-dimensional interleaved array. We need to decimate the array to get the log mag and phase data into two separate arrays. The code snippet is below.

```
//================================================================================
//
// Title:       Example 5
// Purpose:     Acquire Trace Data
//
// Created on:  11/30/2011 by David Judge.
// Copyright:   Anritsu. All Rights Reserved.
//
//================================================================================

//================================================================================
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au464x.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au464x_status = checkErr((fCal)), au464x_status < VI_SUCCESS) \
        goto Error; else


int main (int argc, char *argv[])
{
    ViInt32 retCount;
    ViStatus status;
    ViUInt32 read_count;
    ViStatus au464x_status = VI_SUCCESS;
    FILE* fp;
    ViChar readBuffer[500000];
    ViReal64 fdata[201];
    ViReal64 chanData[402];
    ViReal64 lmData[201];
    ViReal64 phaseData[201];
    int i, ii = 0;
    // We configure the Trace to 2 Rows, 1 column with Trace 1 as LogMag and Phase.
```

```
    // We get the Frequency data and then the Formatted Data.
    // Note that we have 201 data points but we get 402 pieces of data.
    // The data is interleaved so we decimate it in this loop.

    CHECKERR(au464x_init ("VectorStar_Test", VI_FALSE, VI_FALSE, &session));
    CHECKERR(au464x_SetSweepPoints (session, 1, 201));
    CHECKERR(au464x_SetTraceCount (session, 1, 2));
    CHECKERR(au464x_SetTraceFormat (session, 1, 1, AU464X_TRACEFORMAT_LOGPH));
    CHECKERR(au464x_SetTracesLayout (session, 1, AU464X_DISPLAYLAYOUT_R2C1));
    CHECKERR(au464x_GetFrequencyList (session, 1, fdata, 201 , &retCount));
    CHECKERR(au464x_GetFormattedData (session, 1, chanData, 402,&retCount));

    au464x_close(session);

    fp = fopen(".\\chanData.txt","w") ;

    for (i=0;i<201;i++)
    {
        lmData[i] = chanData[ii++];
        phaseData[i] = chanData[ii++];
        fprintf(fp,"%e %10.4f\t%10.4f\n",fdata[i],lmData[i],phaseData[i]);
    }

    fclose(fp);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViUInt32 read_count;
    ViChar  error_buffer [1024];

    ViInt32 my_error_code = 0;
    ViInt32* error_code = &my_error_code;
    ViUInt16 stb;
```

```
    ViUInt16 VNA_ERROR = 4; //This means there is an error
    ViUInt16 VNA_ERROR_MSG = 16; //This means there is a message in the error
buffer(4+16)



    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set
    if (status < VI_SUCCESS | |  (((stb & VNA_ERROR) > 0) && ((stb & VNA_ERROR_MSG) >
0)))
    {
        au464x_writeInstrData (session, ":SYST:ERR?");
        viRead (session, (ViPBuf)error_message, 256, &read_count);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
    au464x_writeInstrData (session, "*CLS");

    }
    return status;
}
```



7.000000e+04	-0.0021	-0.0967
2.000696e+08	0.0001	-2.3994
4.000693e+08	0.0020	-4.7805
6.000690e+08	0.0069	-7.1695
8.000686e+08	0.0020	-9.5621
1.000068e+09	0.0027	-11.9797
1.200068e+09	0.0020	-14.3711
1.400068e+09	0.0017	-16.7732
1.600067e+09	0.0022	-19.1597
1.800067e+09	0.0001	-21.5562
2.000066e+09	0.0045	-23.9807
2.200066e+09	0.0019	-26.3692
2.400066e+09	0.0029	-28.8039
2.600065e+09	-0.0056	-31.1821
2.800065e+09	0.0022	-33.6176
3.000065e+09	-0.0008	-36.0246
3.200064e+09	0.0015	-38.4598
3.400064e+09	0.0019	-40.8751
3.600064e+09	-0.0004	-43.2935
3.800063e+09	0.0036	-45.7318
4.000063e+09	0.0016	-48.1791
4.200063e+09	0.0005	-50.5593
4.400062e+09	0.0017	-52.9950
4.600062e+09	0.0043	-55.4418
4.800062e+09	0.0008	-57.8664
5.000061e+09	-0.0004	-60.2809
5.200061e+09	0.0023	-62.6957
5.400061e+09	0.0070	-65.1525
5.600060e+09	-0.0016	-67.6290
5.800060e+09	-0.0032	-70.0040
6.000060e+09	-0.0026	-72.4109
6.200059e+09	0.0041	-74.8758
6.400059e+09	0.0014	-77.3290

Figure 31—We print out the data into a 3 column format: frequency (Hz), Log Mag, Phase.

Figure B-31.

B-11 Example 6 and 6a– Smith Chart Data

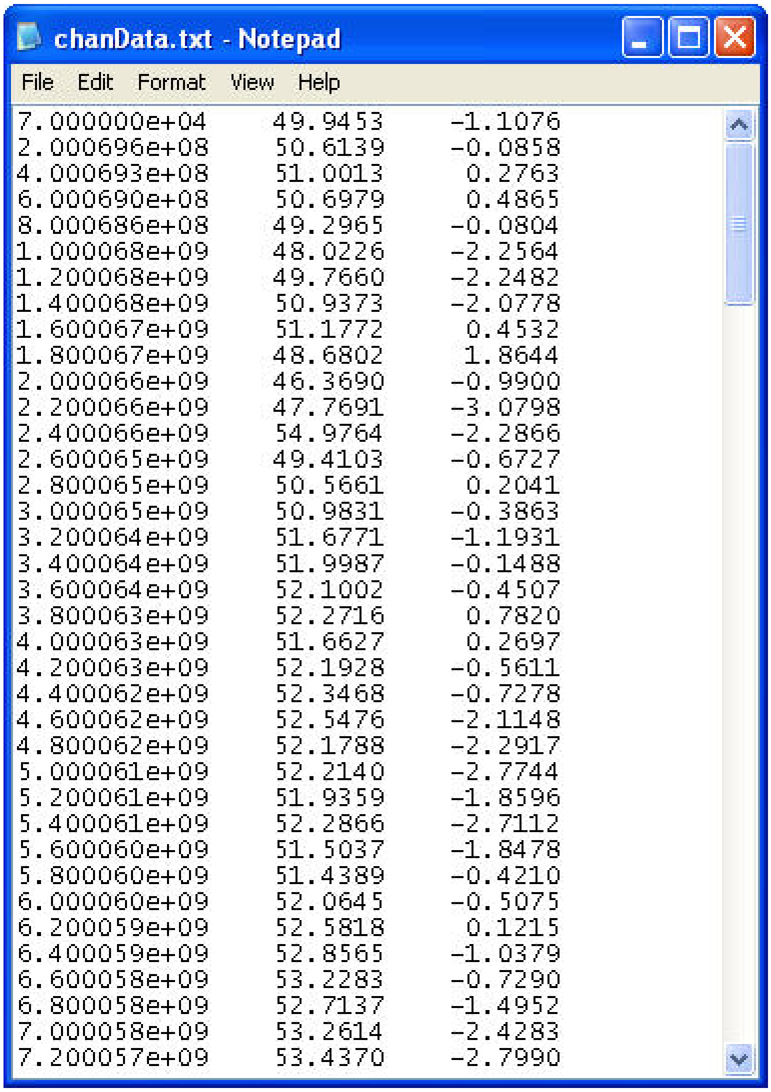
Example 6 – Acquire Smith Chart Trace Data

This example is similar to the previous example except here we do an instrument reset which sets up Smith Charts on Trace 1 and 4. The code snippet is below.

```
//================================================================================
//
// Title:        Example 6
// Purpose:      Acquire Trace Data
//
// Created on:   11/30/2011 by David Judge.
// Copyright:    Anritsu. All Rights Reserved.
//
//================================================================================

//================================================================================
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au464x.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au464x_status = checkErr((fCal)), au464x_status < VI_SUCCESS) \
        goto Error; else


int main (int argc, char *argv[])
{
    ViInt32 retCount;
    ViStatus status;
    ViUInt32 read_count;
    ViStatus au464x_status = VI_SUCCESS;
    FILE* fp;
    ViChar readBuffer[500000];
    ViReal64 fdata[201];
    ViReal64 chanData[402];
    ViReal64 impData[201];
    ViReal64 reactData[201];
    int i, ii = 0;
```

```
    CHECKERR(au464x_init ("VectorStar_Test", VI_FALSE, VI_TRUE, &session));

    CHECKERR(au464x_SetSweepPoints (session, 1, 201));
    CHECKERR(au464x_SetTraceCount (session, 1, 1));
    CHECKERR(au464x_SetTraceFormat (session, 1, 1, AU464X_TRACEFORMAT_SMIT));
    CHECKERR(au464x_SetTracesLayout (session, 1, AU464X_DISPLAYLAYOUT_R1C1));
    CHECKERR(au464x_GetFrequencyList (session, 1, fdata, 201 , &retCount));
    CHECKERR(au464x_GetFormattedData (session, 1, chanData, 402,&retCount));

    au464x_close(session);

    fp = fopen(".\\chanData.txt","w") ;

    for (i=0;i<201;i++)
    {
        impData[i] = chanData[ii++];
        reactData[i] = chanData[ii++];
        fprintf(fp,"%e %10.4f\t%10.4f\n",fdata[i],impData[i],reactData[i]);
    }

    fclose(fp);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;

}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViUInt32 read_count;
    ViChar  error_buffer [1024];



    ViInt32 my_error_code = 0;
    ViInt32* error_code = &my_error_code;
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4; //This means there is an error
```

```
    ViUInt16 VNA_ERROR_MSG = 16; //This means there is a message in the error
buffer(4+16)



    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set
    if (status < VI_SUCCESS | |  (((stb & VNA_ERROR) > 0) && ((stb & VNA_ERROR_MSG) >
0)))
    {
        au464x_writeInstrData (session, ":SYST:ERR?");
        viRead (session, (ViPBuf)error_message, 256, &read_count);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
    au464x_writeInstrData (session, "*CLS");

    }
    return status;
}
```



chanData.txt - Notepad

File Edit Format View Help

7.000000e+04	49.9453	-1.1076
2.000696e+08	50.6139	-0.0858
4.000693e+08	51.0013	0.2763
6.000690e+08	50.6979	0.4865
8.000686e+08	49.2965	-0.0804
1.000068e+09	48.0226	-2.2564
1.200068e+09	49.7660	-2.2482
1.400068e+09	50.9373	-2.0778
1.600067e+09	51.1772	0.4532
1.800067e+09	48.6802	1.8644
2.000066e+09	46.3690	-0.9900
2.200066e+09	47.7691	-3.0798
2.400066e+09	54.9764	-2.2866
2.600065e+09	49.4103	-0.6727
2.800065e+09	50.5661	0.2041
3.000065e+09	50.9831	-0.3863
3.200064e+09	51.6771	-1.1931
3.400064e+09	51.9987	-0.1488
3.600064e+09	52.1002	-0.4507
3.800063e+09	52.2716	0.7820
4.000063e+09	51.6627	0.2697
4.200063e+09	52.1928	-0.5611
4.400062e+09	52.3468	-0.7278
4.600062e+09	52.5476	-2.1148
4.800062e+09	52.1788	-2.2917
5.000061e+09	52.2140	-2.7744
5.200061e+09	51.9359	-1.8596
5.400061e+09	52.2866	-2.7112
5.600060e+09	51.5037	-1.8478
5.800060e+09	51.4389	-0.4210
6.000060e+09	52.0645	-0.5075
6.200059e+09	52.5818	0.1215
6.400059e+09	52.8565	-1.0379
6.600058e+09	53.2283	-0.7290
6.800058e+09	52.7137	-1.4952
7.000058e+09	53.2614	-2.4283
7.200057e+09	53.4370	-2.7990

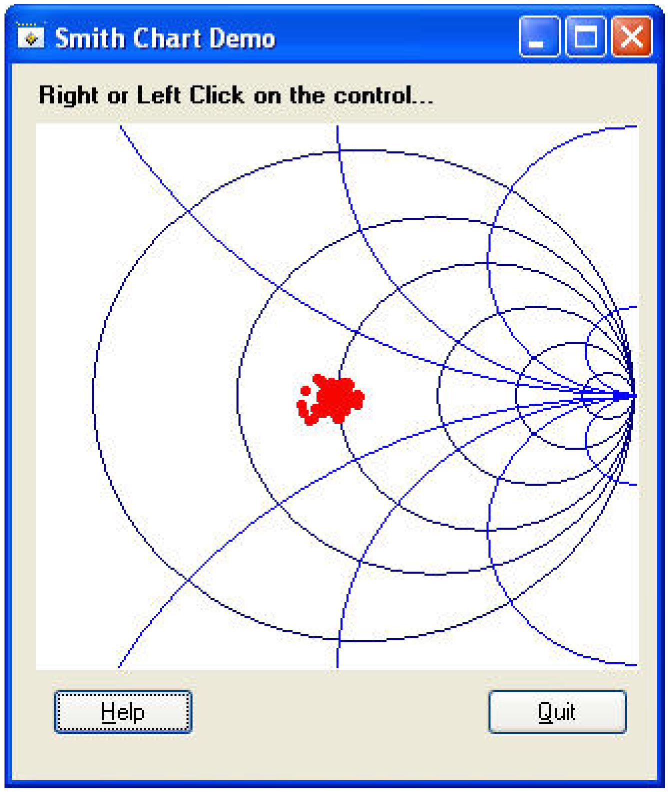
Here is the impedance/reactance data from Trace 1 (Smith Chart). Again we create 3 columns: frequency (Hz), Impedance, Reactance.

Figure B-32. Example 6 – Trace 1 Impedance and Reactance Data

Example 6a – Smith Chart Output

Next we modify Example 6 to use the built-in smith chart control. This program is adapted from the Smith Chart Demo in samples\apps\smithchart. We use the smithchart fp found in toolslib\toolbox\smith.fp." We set Trace 1 to Smith Chart and to output impedance values. Most Smith chart controls actually take a normalized impedance (normalized to 1) so we divide the impedance/reactance pairs by 50 (ohms) to get normalized smith chart data. The code snippet is below.

```
ViInt32 retCount;
ViStatus status;
ViUInt32 read_count;
ViStatus au464x_status = VI_SUCCESS;

ViReal64 fdata[201];
ViReal64 chanData[402];
ViReal64 impData[201];
ViReal64 reactData[201];
int i, ii = 0;

CHECKERR(au464x_init ("VectorStar_Test", VI_FALSE, VI_TRUE, &session));

CHECKERR(au464x_SetSweepPoints (session, 1, 201));
CHECKERR(au464x_SetTraceCount (session, 1, 1));
CHECKERR(au464x_SetTraceFormat (session, 1, 1, AU464X_TRACEFORMAT_SMIT));
CHECKERR(au464x_SetTracesLayout (session, 1, AU464X_DISPLAYLAYOUT_R1C1));
CHECKERR(au464x_GetFrequencyList (session, 1, fdata, 201 , &retCount));
CHECKERR(au464x_GetFormattedData (session, 1, chanData, 402,&retCount));

au464x_close(session);


for (i=0;i<201;i++)
{
    impData[i] = chanData[ii++];
    reactData[i] = chanData[ii++];
    gZ.Real = impData[i]/50.0;
    gZ.Im = reactData[i]/50.0;

    SMITH_PlotImpedancePoint(panelHandle, PANEL_GRAPH, &gZ, VAL_SOLID_CIRCLE,
                             IMPEDANCE_COLOR);
}
```



Use the LabWindows/CVI smith.fp to plot smith chart data.

Figure B-33. Example 6a – Smith Chart Output

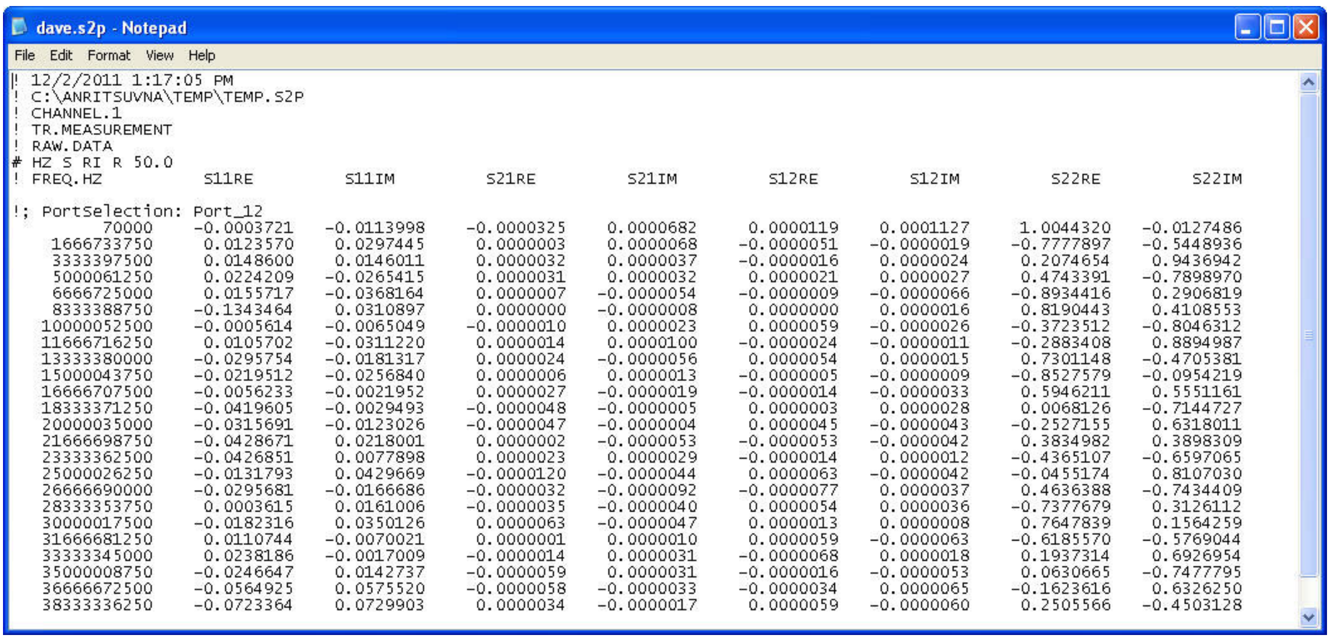
B-12 Example 7 – Output S2P File

This is a good example that uses some au464x_writeInstrData() calls to accomplish some useful things. Specifically, we are going to send some Native VectorStar commands along with some Lightning commands to output an S2P file from the VNA to the PC. The code snippet is below.

```
//=================================================================================
//
// Title:       Example 7
// Purpose:     Output S2P file
//
// Created on:  11/30/2011 by David Judge.
// Copyright:   Anritsu. All Rights Reserved.
//
//=================================================================================
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au464x.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au464x_status = checkErr((fCal)), au464x_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{
    ViInt32 retCount;

    ViStatus status;
    ViUInt32 read_count;
    ViStatus au464x_status = VI_SUCCESS;
    int fileHandle;
    ViChar readBuffer[100000];

    CHECKERR(au464x_init ("VectorStar_Test", VI_FALSE, VI_TRUE, &session));
    CHECKERR(au464x_writeInstrData(session,"LANG NATIVE"));
    CHECKERR(au464x_writeInstrData(session,":SENSE:SWEEP:POINTS 25"));
    CHECKERR(au464x_writeInstrData(session,":FORM:SNP:FREQ HZ"));
    CHECKERR(au464x_writeInstrData(session,":FORM:SNP:PAR REIM"));
    CHECKERR(au464x_writeInstrData(session,"TRS;WFS;OS2P"));
```

```
    CHECKERR(au464x_readArbitraryBlock(session, readBuffer, 1, 100000, VI_TRUE,
                                       &retCount));

    au464x_close(session);

    fileHandle = OpenFile (".\\dave.s2p", VAL_WRITE_ONLY, VAL_OPEN_AS_IS, VAL_ASCII);
    WriteFile (fileHandle, readBuffer, retCount);
    CloseFile (fileHandle);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViUInt32 read_count;
    ViChar  error_buffer [1024];



    ViInt32 my_error_code = 0;
    ViInt32* error_code = &my_error_code;
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4; //This means there is an error
    ViUInt16 VNA_ERROR_MSG = 16; //This means there is a message in the error
buffer(4+16)



    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set
    if (status < VI_SUCCESS ||  (((stb & VNA_ERROR) > 0) && ((stb & VNA_ERROR_MSG) >
0)))
    {
        au464x_writeInstrData (session, ":SYST:ERR?");
        viRead (session, (ViPBuf)error_message, 256, &read_count);
        SetWaitCursor (0);
```

```
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
    au464x_writeInstrData (session, "*CLS");

    }
    return status;
}
```




```
! 12/2/2011 1:17:05 PM
! C:\ANRITSUVNA\TEMP\TEMP.S2P
! CHANNEL.1
! TR.MEASUREMENT
! RAW.DATA
# HZ S RI R 50.0
! FREQ.HZ          S11RE        S11IM        S21RE        S21IM        S12RE        S12IM        S22RE        S22IM

!; PortSelection: Port_12
        70000  -0.0003721  -0.0113998  -0.0000325   0.0000682   0.0000119   0.0001127   1.0044320  -0.0127486
   1666733750   0.0123570   0.0297445   0.0000003   0.0000068  -0.0000051  -0.0000019  -0.7777897  -0.5448936
   3333397500   0.0148600   0.0146011   0.0000032   0.0000037  -0.0000016   0.0000024   0.2074654   0.9436942
   5000061250   0.0224209  -0.0265415   0.0000031   0.0000032   0.0000021   0.0000027   0.4743391  -0.7898970
   6666725000   0.0155717  -0.0368164   0.0000007  -0.0000054  -0.0000009  -0.0000066  -0.8934416   0.2906819
   8333388750  -0.1343464   0.0310897   0.0000000  -0.0000008   0.0000000   0.0000016   0.8190443   0.4108553
  10000052500  -0.0005614  -0.0065049  -0.0000010   0.0000023   0.0000059  -0.0000026  -0.3723512  -0.8046312
  11666716250   0.0105702  -0.0311220   0.0000014   0.0000100  -0.0000024  -0.0000011  -0.2883408   0.8894987
  13333380000  -0.0295754  -0.0181317   0.0000024  -0.0000056   0.0000054   0.0000015   0.7301148  -0.4705381
  15000043750  -0.0219512  -0.0256840   0.0000006   0.0000013  -0.0000005  -0.0000009  -0.8527579  -0.0954219
  16666707500  -0.0056233  -0.0021952   0.0000027  -0.0000019  -0.0000014  -0.0000033   0.5946211   0.5551161
  18333371250  -0.0419605  -0.0029493  -0.0000048  -0.0000005   0.0000003   0.0000028   0.0068126  -0.7144727
  20000035000  -0.0315691  -0.0123026  -0.0000047  -0.0000004   0.0000045  -0.0000043  -0.2527155   0.6318011
  21666698750  -0.0428671   0.0218001   0.0000002  -0.0000053  -0.0000053  -0.0000042   0.3834982   0.3898309
  23333362500  -0.0426851   0.0077898   0.0000023   0.0000029  -0.0000014   0.0000012  -0.4365107  -0.6597065
  25000026250  -0.0131793   0.0429669  -0.0000120  -0.0000044   0.0000063  -0.0000042  -0.0455174   0.8107030
  26666690000  -0.0295681  -0.0166686  -0.0000032  -0.0000092  -0.0000077   0.0000037   0.4636388  -0.7434409
  28333353750   0.0003615   0.0161006  -0.0000035  -0.0000040   0.0000054   0.0000036  -0.7377679   0.3126112
  30000017500  -0.0182316   0.0350126   0.0000063  -0.0000047   0.0000013   0.0000008   0.7647839   0.1564259
  31666681250   0.0110744  -0.0070021   0.0000001   0.0000010   0.0000059  -0.0000063  -0.6185570  -0.5769044
  33333345000   0.0238186  -0.0017009  -0.0000014   0.0000031  -0.0000068   0.0000018   0.1937314   0.6926954
  35000008750  -0.0246647   0.0142737  -0.0000059   0.0000031  -0.0000016  -0.0000053   0.0630665  -0.7477795
  36666672500  -0.0564925   0.0575520  -0.0000058  -0.0000033  -0.0000034   0.0000065  -0.1623616   0.6326250
  38333336250  -0.0723364   0.0729903   0.0000034  -0.0000017   0.0000059  -0.0000060   0.2505566  -0.4503128
```

Transfer of an S2P file to the PC.

Figure B-34. Example 7 – S2P Output File

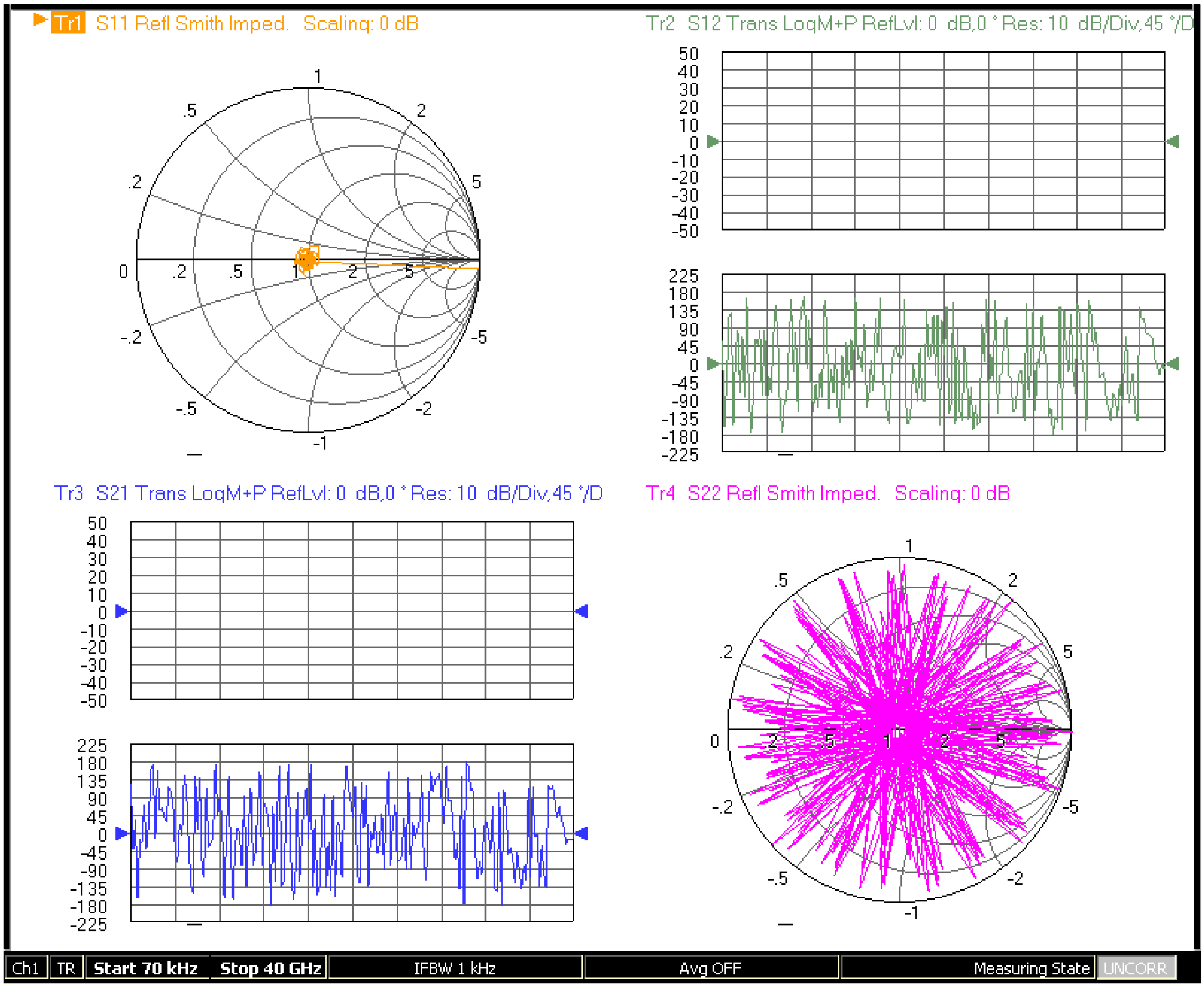
B-13 Example 8 – Output BMP File

We can use a similar technique to get the bitmap data to a file. Here we use the Lightning commands “BMPC;OBMP” to output a bitmap file. BMPC selects color on white as the color scheme – this makes for better printouts. The au464x_readArbitraryBlock() function again is used to strip off the arbitrary block header and place the bitmap data into a file. If we didn’t strip off the header the bitmap file would be corrupt. The code snippet is below.

```
//==================================================================================
//
// Title:       Example 8
// Purpose:     Output BMP file
//
// Created on:  11/30/2011 by David Judge.
// Copyright:   Anritsu. All Rights Reserved.
//
//==================================================================================

//==================================================================================
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au464x.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au464x_status = checkErr((fCal)), au464x_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{
    ViInt32 retCount;
    ViStatus status;
    ViUInt32 read_count;
    ViStatus au464x_status = VI_SUCCESS;
    int fileHandle;
    static ViChar readBuffer[600000];

    CHECKERR(au464x_init ("VectorStar_Test", VI_FALSE, VI_TRUE, &session));
    CHECKERR(au464x_writeInstrData(session,"LANG NATIVE"));
    CHECKERR(au464x_writeInstrData(session,"BMPC;OBMP"));

    CHECKERR(au464x_readArbitraryBlock(session, readBuffer, 1, 600000, VI_TRUE,
&retCount));
```

```
    au464x_close(session);

    fileHandle = OpenFile (".\\dave.bmp", VAL_WRITE_ONLY, VAL_OPEN_AS_IS, VAL_ASCII);
    WriteFile (fileHandle, readBuffer, retCount);
    CloseFile (fileHandle);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViUInt32 read_count;
    ViChar  error_buffer [1024];

    ViInt32 my_error_code = 0;
    ViInt32* error_code = &my_error_code;
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4; //This means there is an error
    ViUInt16 VNA_ERROR_MSG = 16; //This means there is a message in the error
buffer(4+16)

    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set
    if (status < VI_SUCCESS | |  (((stb & VNA_ERROR) > 0) && ((stb & VNA_ERROR_MSG) >
0)))
    {
        au464x_writeInstrData (session, ":SYST:ERR?");
        viRead (session, (ViPBuf)error_message, 256, &read_count);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
    au464x_writeInstrData (session, "*CLS");

    }
```

```
    return status;
}
```



When the VI runs it puts up a dialog to allow the user to select a file name. Make sure to save the file with a ".bmp" extension.

Figure B-35. Example 8 – Trace Display Output Saved as a BMP File

Appendix C — Programming with VISA/C#

C-1 Introduction

This document provides an overview of programming techniques for controlling the VectorStar VNA using the NI-VISA driver and C# over GPIB, TCP/IP (using VXI-11), and USB. Programming using TCP/IP Sockets requires some special handling and is not discussed in this document.

C-2 Overview

Programming Basics

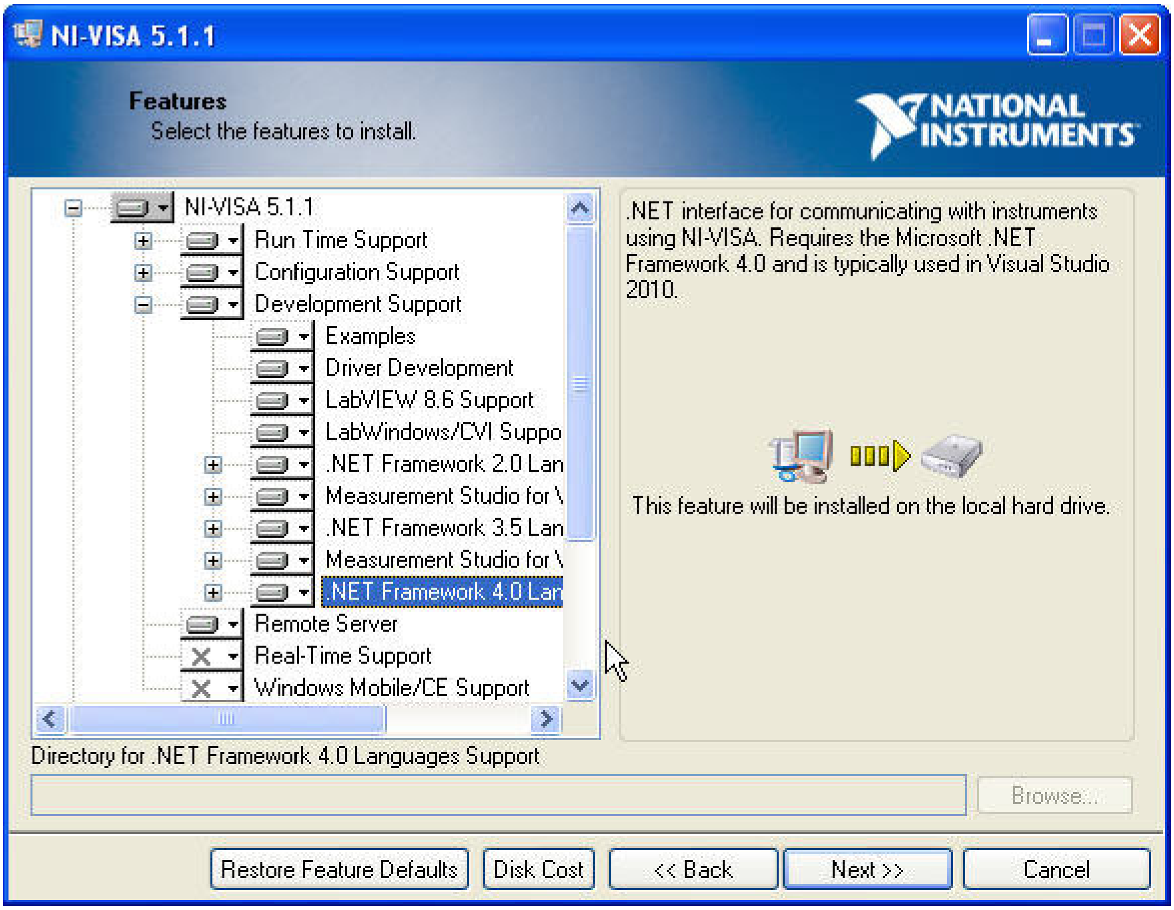
There are many cases where a user may want programmatic control of the VNA. Examples include automating a test sequence, orchestrating a complex measurement involving various pieces of test equipment, gathering a time series of data, or as a convenient way of getting data or images transferred from the VNA to a PC for further analysis.

What is VISA?

VISA (Virtual Instrument System Architecture) is an I/O software standard for communicating with test instruments like VectorStar over any of the bus architectures which VectorStar supports. A VISA driver is available from both National Instruments and Agilent. National Instruments VISA drivers are available for the following operating systems: Windows (all versions), Mac OS X, Linux, Mandriva, RedHat, SUSE, PharLap, VxWorks, Pocket PC 2003, and Windows CE/Mobile. It's always a good idea to get the latest driver (Version 5.1.1), but make sure to get the Full Version (not just the runtime) version 5.0 or higher for the best support of the latest .NET 4.0 Framework, USB and TCP/IP. The driver is available from http://www.ni.com/visa/ or from NI Device Driver CD that comes with NI hardware.

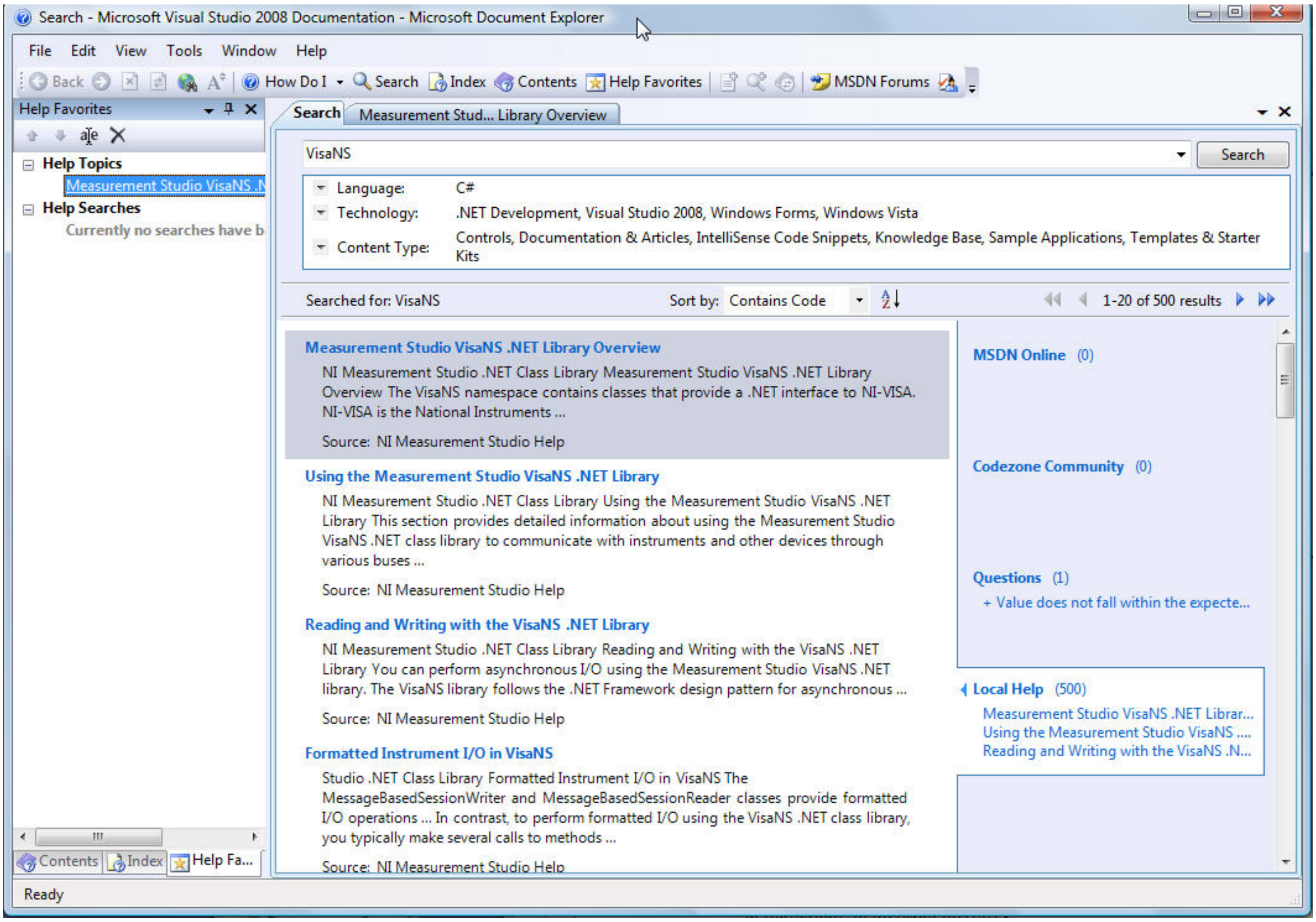
Installing NI-VISA 5.1.1 Driver.

Figure C-1. Install NI-VISA Driver



From the Full NI-VISA 5.1.1 installation, you can also select to install Developer Support for .NET Frameworks 2.0, 3.5 and 4.0. It does not automatically install Development Support for the .NET Frameworks so you have to select these to install. Make sure to select “Examples” also. The VISA Version 5.1.1 driver supports the .NET Framework 2.0, 3.5 and 4.0.

Figure C-2. NI-VISA Installation



Search for “VisaNS” in Visual Studio (Local) help to find more information about using the VISA driver.

Figure C-3. VisaNS Search

Programming Environments

Programming environments that are commonly used with test equipment include LabVIEW, LabWindows/CVI, Microsoft Visual Studio (2005, 2008, 2010), Visual Basic 6, HP Basic, and so on. NI-VISA supports the following Windows development environments:

- LabVIEW
- LabWindows/CVI
- Measurement Studio for Visual Studio
- Microsoft Visual Studio 2010/2008/2005/2003/6.0
- Microsoft Visual Basic 6.0

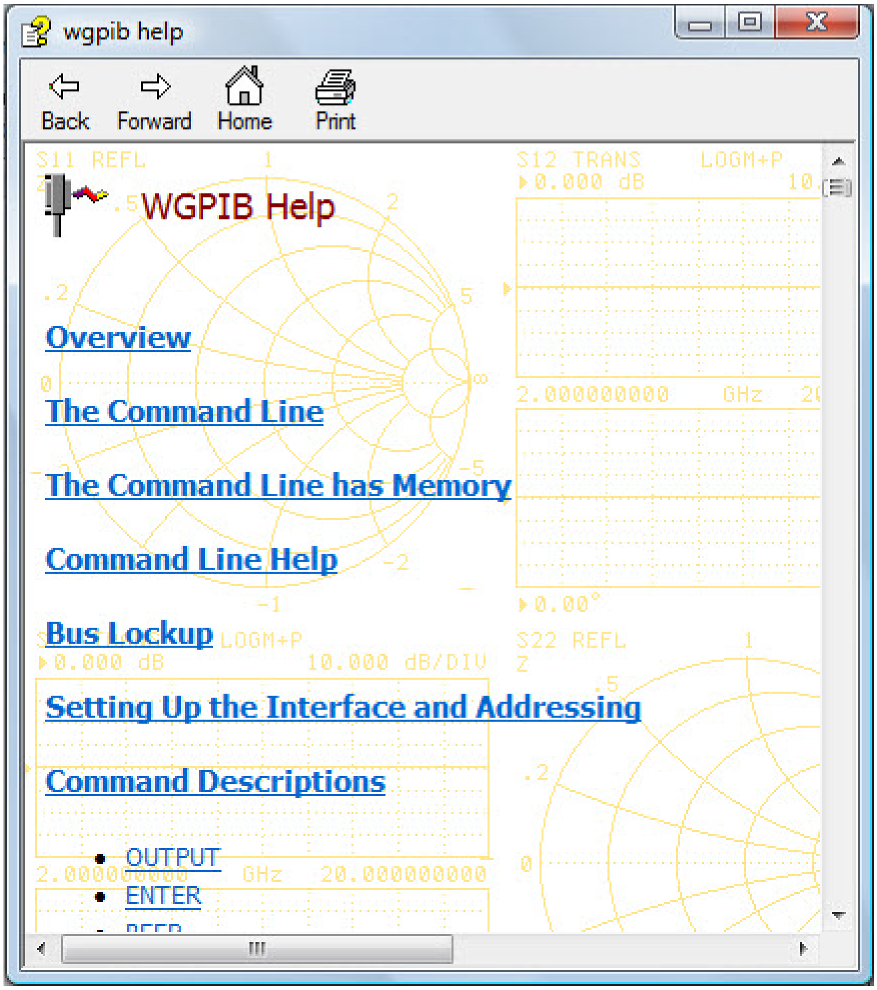
For the examples below, we’ll use Visual C# 2010 Express Edition. You can download the latest Express Edition for free from the Microsoft website http://www.microsoft.com/express/download.

GPIB Mnemonics

VectorStar has a fairly large set of GPIB commands. This includes a set of Native commands, Lightning commands and Agilent 8510 commands. You use the same commands regardless of the communication method employed. See Example 3 for using the LIST command to output the full set of supported GPIB mnemonics directly from the VNA.

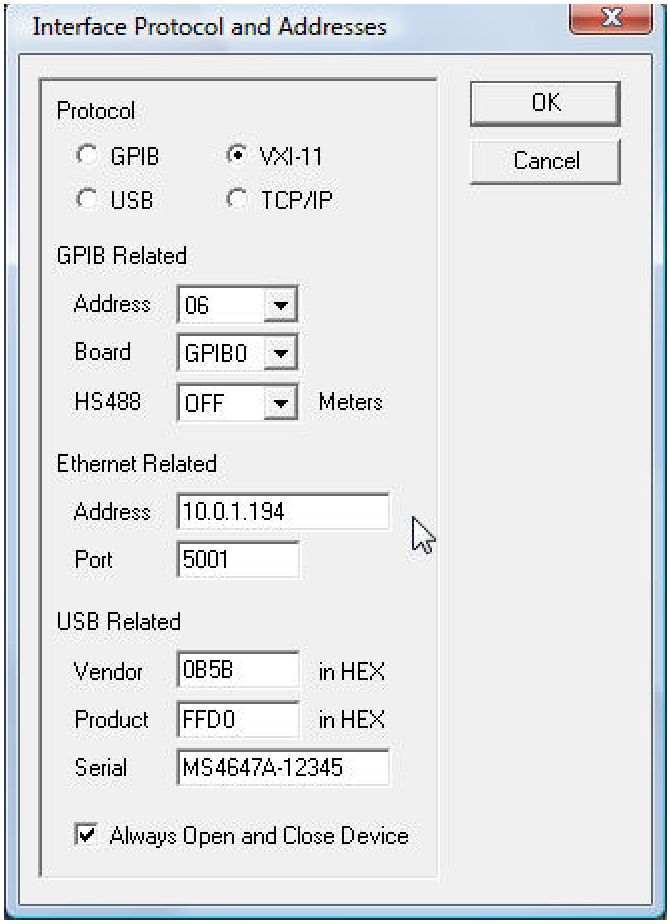
C-3 A Quick Start with the Windows GPIB Utility

Windows GPIB (WGPIB) is a useful windows program that is available from the Anritsu web site. The latest version is 3.75 and you'll want to get that one or higher to get the latest communication functions. Before writing any software, make sure you can write to and read from VectorStar over the communication protocol you're interested in.



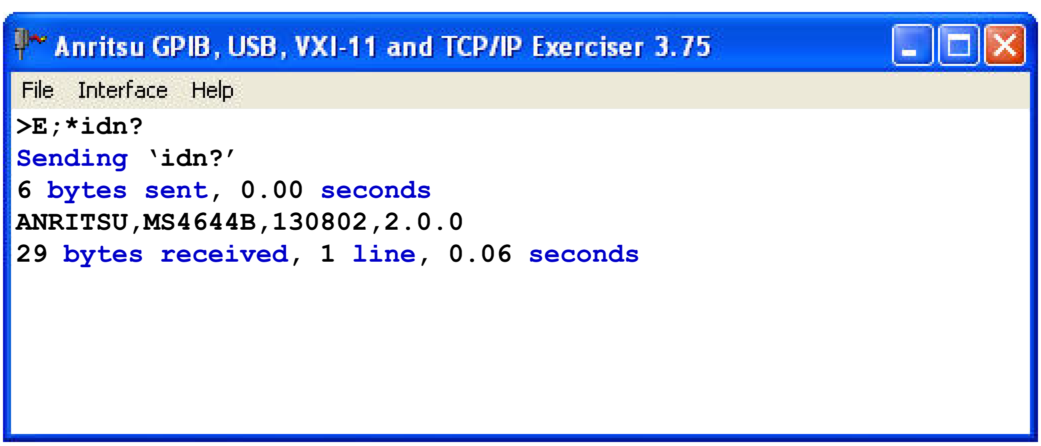
Take a quick look at WGPIB Help to get an overview of the application.

Figure C-4. WGPIB Help



Example of setting up a VXI-11 connection to VectorStar. The VectorStar TCP/IP address is 10.0.1.194 in this example.

Figure C-5. VXI-11 TCP/IP Connection



The result of using WGPIB to query the instrument using the "*IDN?" command.

Figure C-6. WGPIB Online Help

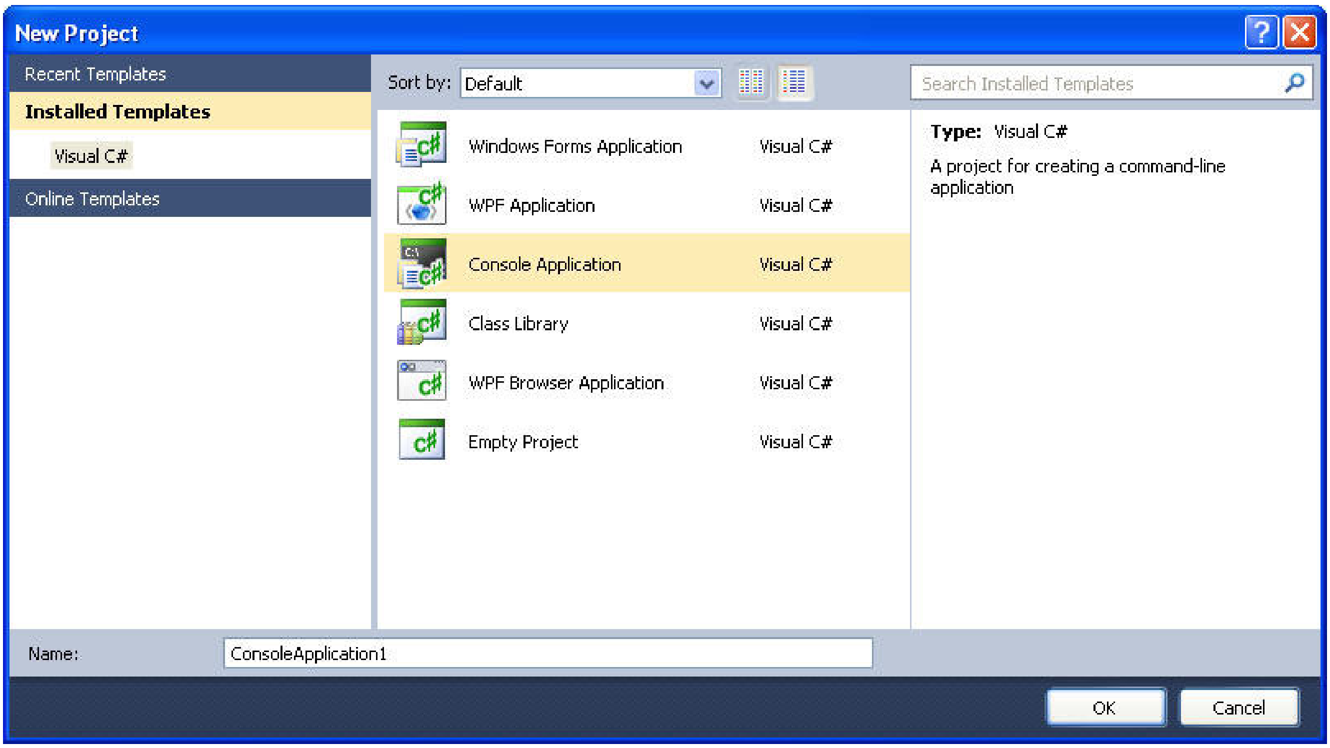
References

- NI-VISA User Manual
- NI-VISA Programmer Reference Manual
- Lightning Programming Manual – 10410-00262
- VectorStar Programming Manual – IEEE, System, and SCPI Commands – 10410-00322
- VectorStar Programming Manual Supplement – Lightning 37xxxx and HP8510 Commands – 10410-00323

C-4 Programming Examples

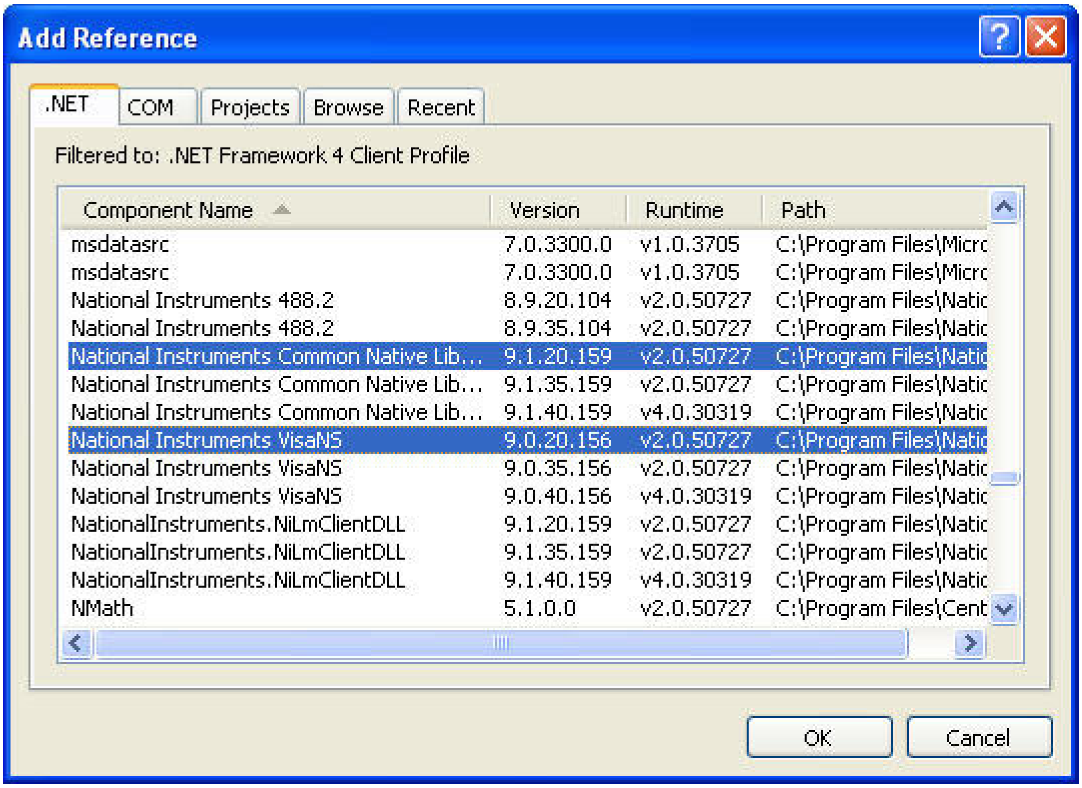
Getting Started

We'll create a few simple Console-based programs to demonstrate the use of the NI-VISA driver for controlling VectorStar.



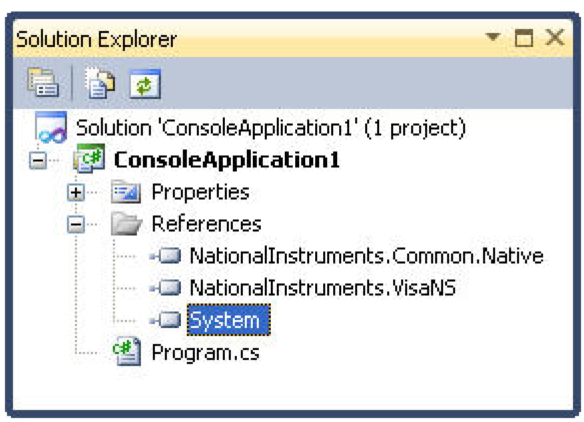
Create a new Console Application Project.

Figure C-7. Getting Started – Console Application Project



Add references for National Instruments Common and National Instruments VisaNS. This shows the .NET 2.0 versions of these DLLs.

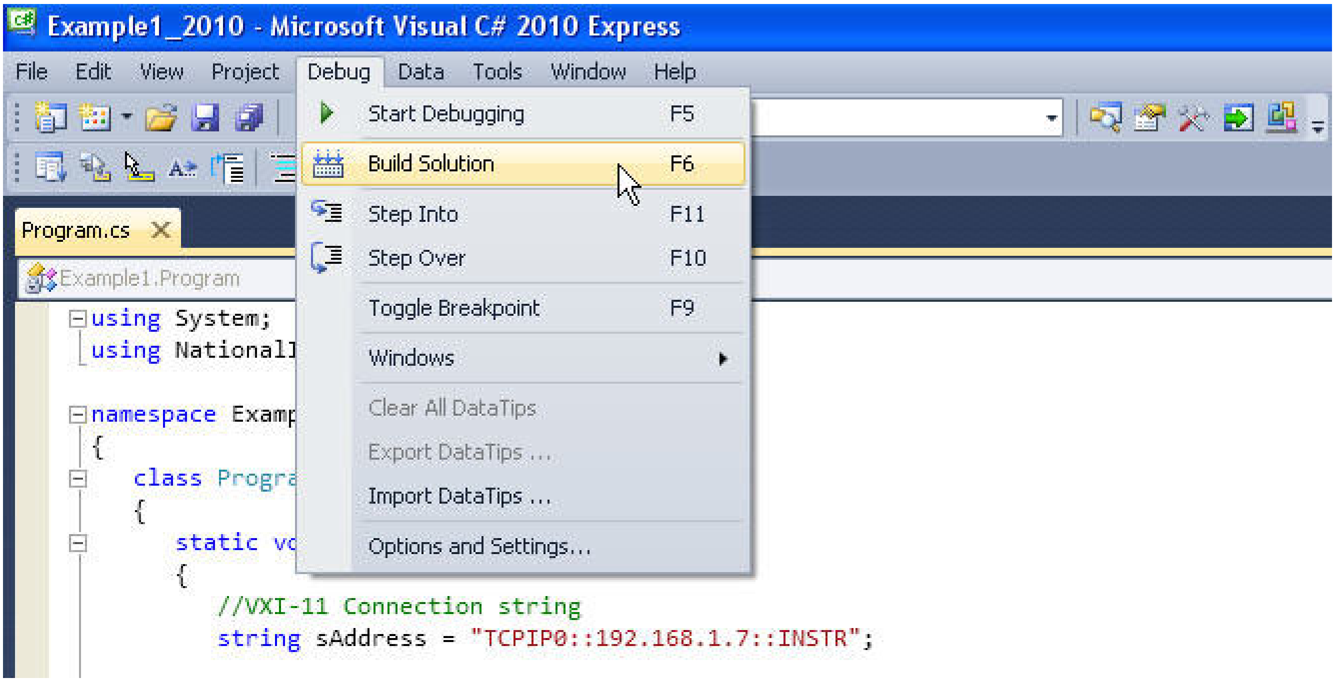
Figure C-8. Getting Started – References to VisaNS



For simple programs (like all the examples below) only these three references are required.

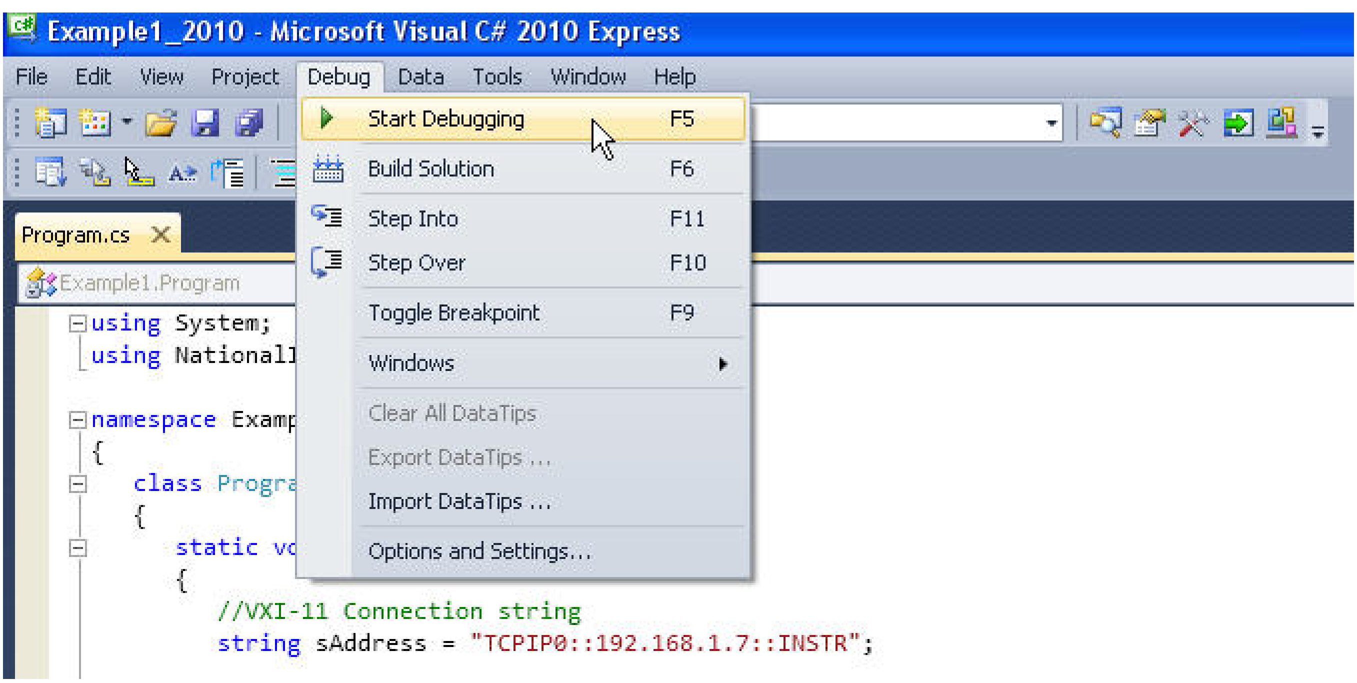
Figure C-9. Getting Started – Solution Explorer References

If you have the references set up correctly you can copy and paste the code into the Program.cs file. Then Build the solution and run (Start Debugging). Each example includes the complete program code.



Build the solution.

Figure C-10. Getting Stgarted – Building the Solution



Start Debugging.

Figure C-11. Getting Started – Debugging

C-5 Example 1 — Open Session and *IDN?

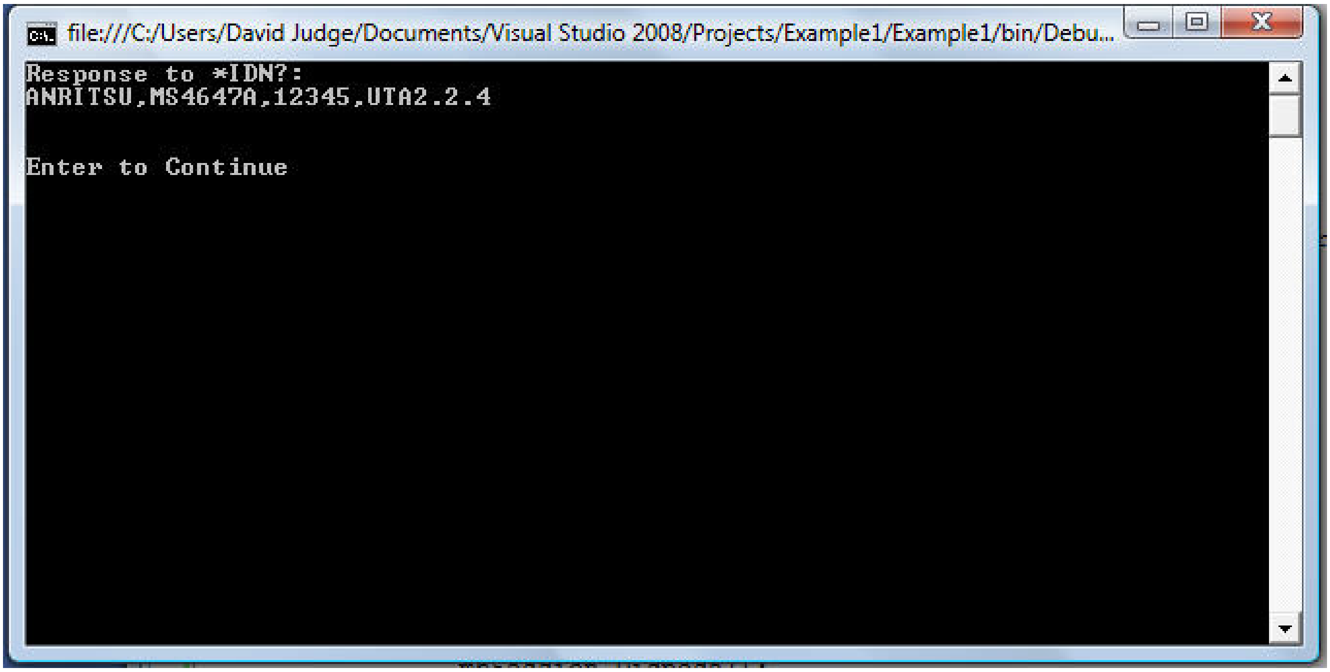
Example 1 – Code Listing

```
using System;
using NationalInstruments.VisaNS;

namespace Example1
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";

            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;

            //response string
            string responseString = null;

            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Send "*IDN?" command
                mbSession.Write("*IDN?\n");
                //Notice that the Session Write commands are terminated with a "/n".
                //This is the newline character and is not really needed for GPIB,
                //USB or VXI-11, but is needed for Sockets.
                //It doesn't cause any problems to include it.
                //Read the response
                responseString = mbSession.ReadString();

                //Write to Console
                Console.WriteLine("Response to *IDN?:");
```

```
                Console.WriteLine(responseString);
                //Return to Local Control
                mbSession.Write("RTL\n");
                //Close the Session
                mbSession.Dispose();

            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
                Console.WriteLine();
            }

            keepConsoleUp();
            // This is here so the Console stays up until the user hits return.
        }

        private static void keepConsoleUp()
        {
            Console.WriteLine("");
            Console.WriteLine("Enter to Continue");
            Console.ReadLine();
        }
    }
}
```

Example 1 – Discussion

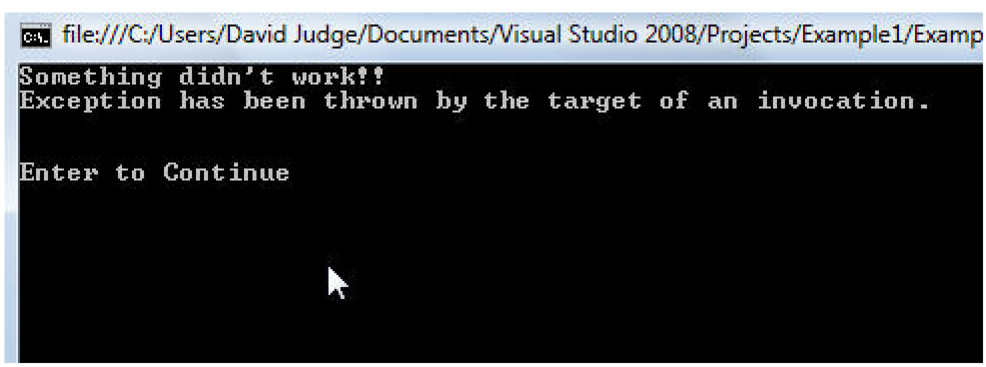
1. The program output shown below.



The console output should look something like this if everything worked. The result is Manufacturer, Model, Serial Number, and Firmware Version.

Figure C-12. Example 1 – Console Output

2. If it didn't work then check that the sAddress string matches your setup. This is the error you'll get if the address is wrong as shown in the figure below.



Results if the connection string is no good.

Figure C-13. Example 1 – Exception Message

- The first thing to observe is the VISA connection string. Here are some possible strings:

```
//VXI-11 Connection string
string sAddress = "TCPIP0::10.0.1.194::INSTR";

//GPIB Connection string
string sAddress = "GPIB0::6::INSTR";

//USB Connection string (vendor::product::serial_number)
string sAddress = "USB0::0x0B5B::0xFFD0::MS4647B-12345::INSTR";
```

- The beauty of using VISA is that the only thing that needs to be changed for any of these possible communication protocols is the connection string. The rest of the code should be exactly the same (except for SOCKETS). For TCP/IP we recommend using VXI-11 since it better implements the IEEE 488.2 standard and all status checking.
- But just in case, the connection string for a TCP/IP socket connection is:

```
//TCP/IP Sockets Connection string
string sAddress = "TCPIP0::10.0.1.194::5001::SOCKET";
```

3. Opening the communication session.

```
//open a Session to the VNA
mySession = ResourceManager.GetLocalManager().Open(sAddress);
//cast this to a message based session
mbSession = (MessageBasedSession)mySession;
```

4. Write to VNA and Read from VNA

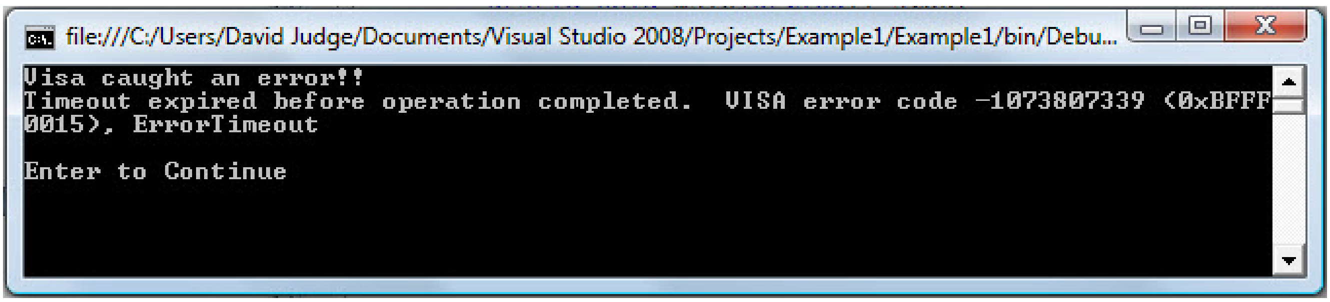
```
//Send "*IDN?" command
mbSession.Write("*IDN?\n");
//Read the response
responseString = mbSession.ReadString();
```

- Notice that the mbSession.Write() commands are terminated with a “/n”. This is the newline character. The next example discusses session parameters. Some of the other examples will get into reading binary data and arbitrary block data.

5. Exception Handling

- The VISA driver catches certain types of errors like timeouts and .NET catches the rest.

```
catch (VisaException v_exp)
{
    Console.WriteLine("Visa caught an error!!");
    Console.WriteLine(v_exp.Message);
}
catch (Exception exp)
{
    Console.WriteLine("Something didn't work!!");
    Console.WriteLine(exp.Message);
    Console.WriteLine();
}
```



VisaException catches a timeout on a ReadString()

Figure C-14. Example 1 – VisaException and Timeout

6. Close the session using Dispose()

```
//Close the Session
mbSession.Dispose();
```

C-6 Example 2 — Session Parameters and Status Checking

Example 2 – Code Listing

```
using System;
using NationalInstruments.VisaNS;
using System.Threading;

namespace Example2
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";

            //The VNA uses a message based session.
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            try
            {
                //Open a Session to the VNA.
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //Cast this to a message based session.
                mbSession = (MessageBasedSession)mySession;

                //Now set some Session Parameters.
                //Timeout to 1 second (1000 ms).
                mbSession.Timeout = 1000;
                //Session Parameters were all set to default values in Example 1.
                //Here we specifically set them.

                //Use newline (\n or 0x0a) as termination character.
                mbSession.TerminationCharacter = 0x0a;
                //Terminate reads if the termination character is spotted.
                mbSession.TerminationCharacterEnabled = false;
                //Assert an END during the transfer of the last byte of data.
                mbSession.SendEndEnabled = true;
```

```
        //Send a *CLS
        testClear(mbSession);

        //Now try a few commands - first *IDN?
        //Note that the \n is not included in these strings
        //LANG NATIVE below allows us to work with the
        //VectorStar Status Register setup in this example.
        mbSession.Write("LANG NATIVE");
        mbSession.Write("*IDN?");
        Console.WriteLine("Response to *IDN?:");
        testStatus_Read(mbSession);

        //OID returns an instrument ID string
        mbSession.Write("OID");
        Console.WriteLine("Response to OID:");
        testStatus_Read(mbSession);

        //*OPT? returns the installed instrument options
        mbSession.Write("*OPT?");
        Console.WriteLine("Response to *OPT?:");
        testStatus_Read(mbSession);

        //This command doesn't exist, check that we catch it
        //This command should be caught in testStatus().
        mbSession.Write("ABC");
        Console.WriteLine("Test ABC:");
        testStatus(mbSession);

        //This command is OK, no response is expected
        mbSession.Write("CH3");
        Console.WriteLine("Test CH3:");
        testStatus(mbSession);
    //Return to Local Control
    mbSession.Write("RTL\n");
    //Close the Session
    mbSession.Dispose();
    }
    catch (VisaException v_exp)
    {
        Console.WriteLine("Visa caught an error!!");
```

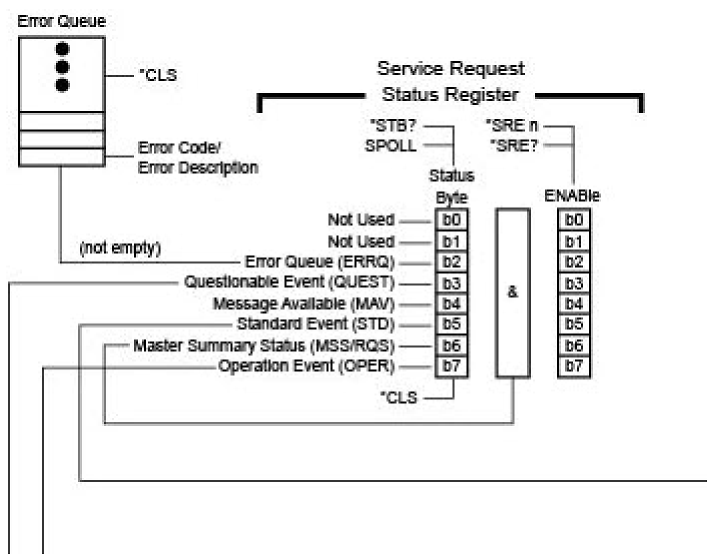
```
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
            }

            keepConsoleUp();
        }

        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
        }

        //This function tests the Service Request Status Register
        //and does a read if MAV is set.
        //It prints an error message if any error is reported.
        private static void testStatus_Read(MessageBasedSession mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available

            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();
            string responseString = null;


            while (((int)sb & (b2 + b4)) == 0)
            //Before we start the read we want to see the MAV bit set (b4).
            //But if an error occurs (b2) we’ll report it.
            //We could add another test to make sure we don’t loop here forever.
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
```

```
            responseString = mbSession.Query("OGE\n");
            Console.WriteLine("Error Queue: " + responseString);

        }
        else if (((int)sb & b4) != 0)
        {
            responseString = mbSession.ReadString();
            Console.WriteLine(responseString);
        }
        mbSession.Write("*CLS\n");
    }

    //This function checks if the GPIB command raises an error.
    //This function tests the Service Request Status Register.
    //It prints an error message if any error is reported.
    private static void testStatus(MessageBasedSession mbSession)
    {
        //These are the bits to check
        int b2 = 4;     //Error Queue is not empty

        //Read the Status Byte of Service Request Status Register
        //wait 50 ms
        Thread.Sleep(50);

        //then check status
        StatusByteFlags sb = mbSession.ReadStatusByte();
        string responseString = null;

        if (((int)sb & b2) != 0)
        {
            responseString = mbSession.Query("OGE\n");
            Console.WriteLine("Error Queue: " + responseString);

        }
        else
        {
            Console.WriteLine("OK");
        }

        mbSession.Write("*CLS\n");
```

```
        }
        private static void keepConsoleUp()
        {
            Console.WriteLine("");
            Console.WriteLine("Enter to Continue");
            Console.ReadLine();
        }

    }
}
```

Example 2 – Discussion

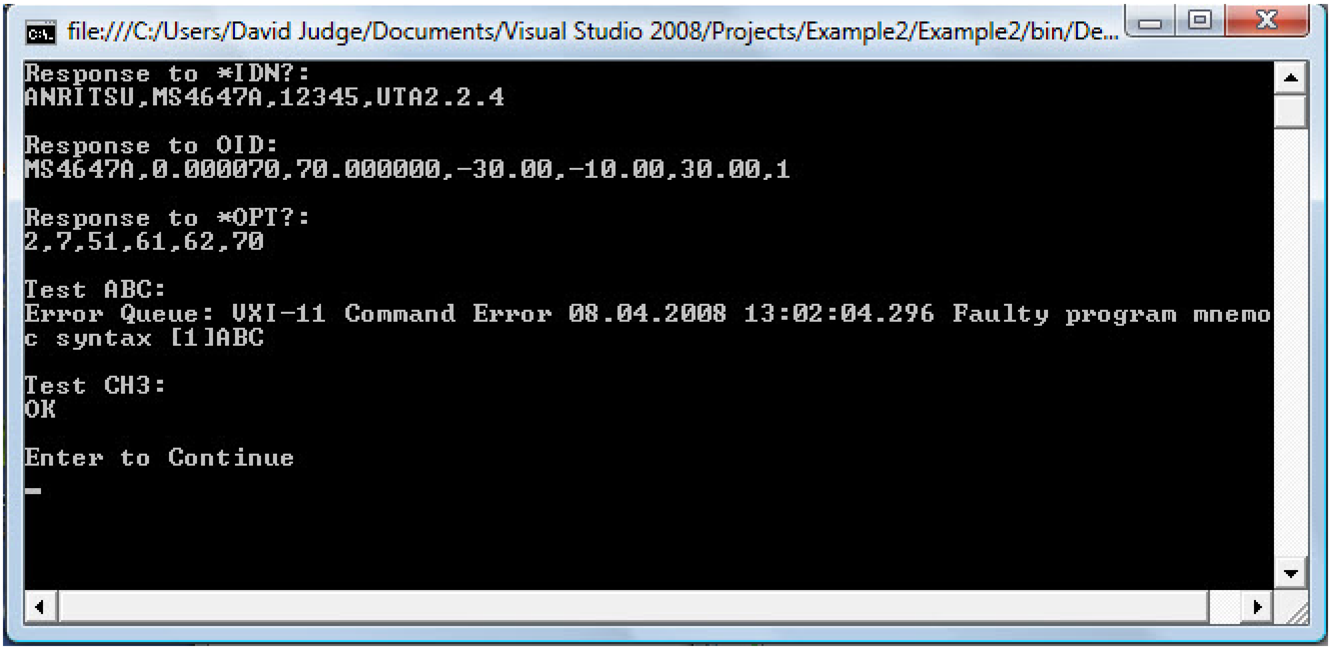
1. In this example we check the Status Byte prior to reading the response.
    - The Status Byte is the Service Request Status Register and is discussed elsewhere.
    - We set the LANG to NATIVE to use the VectorStar Status Register Configuration.
    - In general we expect to get a Message Available (MAV) response when data is available to read.
    - Otherwise we expect something to be in the Error Queue (ERRQ).
    - Notice when the code send an unknown command “ABC” and the testStatus() function catches it and outputs the message in the Error Queue.



The Service Request Status Register. It is slightly changed from Lightning to VectorStar. If LANG LIGHT is set then the Lightning configuration of the Status Register is used.

Figure C-15. Example 2 – Service Request Status Register

2. The expected output is shown below.



Note the error message after sending the “ABC” fake command.

Figure C-16. Example 2 – Faulty Command Message

C-7 Example 3 — LIST Command

Example 3 – Code Listing

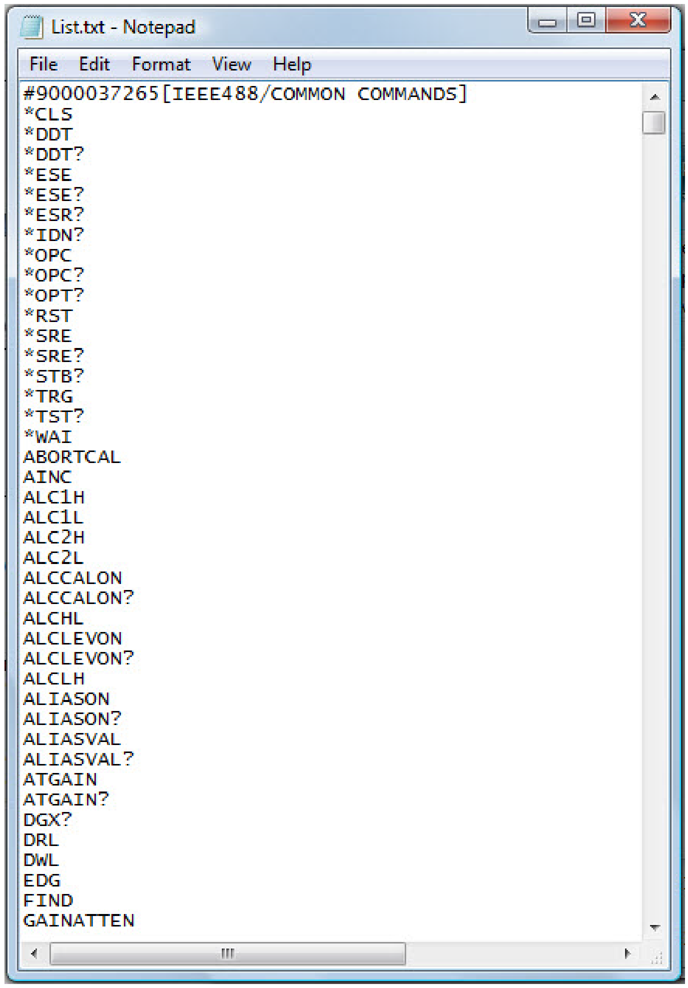
```
using System;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;

namespace Example3
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";

            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;

            string responseString = null;
            string[] split = null;

            try
            {
                //Open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //Cast this to a message based session
                mbSession = (MessageBasedSession)mySession;

                //Timeout to 1 second (1000 ms)
                mbSession.Timeout = 1000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;

                //We're expected a lot of response data to the LIST command
                //We’ll increase the size of the read buffer.
```

```
                mbSession.DefaultBufferSize = 50000;
                testClear(mbSession);

                //Set the Language to NATIVE and send the LIST command
                mbSession.Write("LANG NATIVE\n");
                mbSession.Write("LIST\n");
                responseString = testStatus_Read(mbSession);
                //The results are delimited by a newline (\n)
                //The Split() function is handy for many string parsing functions.
                split = responseString.Split('\n');

                //Send results to a file
                //Results are written to a file in the same directory as the exe.
                StreamWriter output = new StreamWriter("List.txt");
                foreach (string s in split)
                    output.WriteLine(s);
                output.Close();
                mbSession.Write("RTL\n");
                mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }

            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
            }
            keepConsoleUp();
        }

        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
        }
       private static void keepConsoleUp()
        {
```

```
            Console.WriteLine("");
            Console.WriteLine("Enter to Continue");
            Console.ReadLine();
        }
        //This function tests the Service Request Status Register and
        //does a read if MAV is set.
        //It prints an error message if any error is reported.
        private static string testStatus_Read(MessageBasedSession mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available

            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();
            string responseString = null;
            string errorString = null;

            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                errorString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + errorString);

            }
            else if (((int)sb & b4) != 0)
            {
                responseString = mbSession.ReadString();
            }
            mbSession.Write("*CLS\n");
            return responseString;
        }
    }
}
```

Example 3 – Discussion

1. This and the next few examples don't write anything to the Console unless there is an error. Output is sent to file in the same directory of the running program (\Example3\Bin\Debug\List.txt).




```
#9000037265[IEEE488/COMMON COMMANDS]
*CLS
*DDT
*DDT?
*ESE
*ESE?
*ESR?
*IDN?
*OPC
*OPC?
*OPT?
*RST
*SRE
*SRE?
*STB?
*TRG
*TST?
*WAI
ABORTCAL
AINC
ALC1H
ALC1L
ALC2H
ALC2L
ALCCALON
ALCCALON?
ALCHL
ALCLEVON
ALCLEVON?
ALCLH
ALIASON
ALIASON?
ALIASVAL
ALIASVAL?
ATGAIN
ATGAIN?
DGX?
DRL
DWL
EDG
FIND
GAINATTEN
```

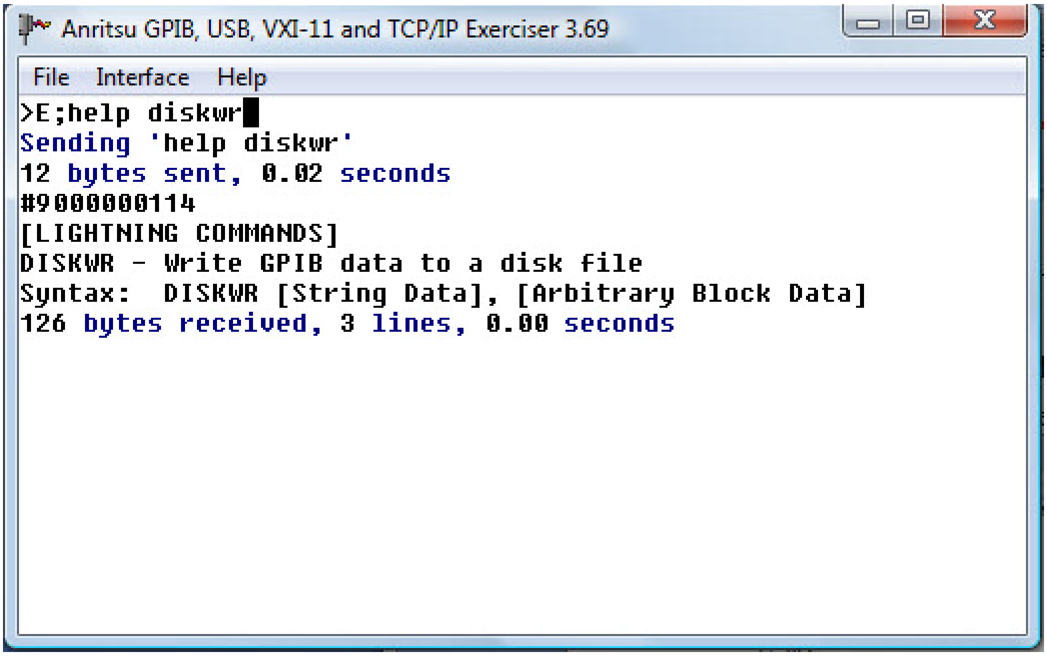
The List of all commands supported by VectorStar.

Figure C-17. Example 3 – VectorStar Command Listing

2. Use LANG LIGHT. Lightning has the same command but is comma delimited:

```
//Set the Language to LIGHTNING and send the LIST command
mbSession.Write("LANG LIGHT\n");
mbSession.Write("LIST\n");
responseString = testStatus_Read(mbSession);
//the results are delimited by a comma
split = responseString.Split(',');
```

3. Use WGPIB to get more help on any command.



Help will tell you what type of command (Lightning, Native, 8510) you're asking about and provides syntax.

Figure C-18. Example 3 – WGPIB Help Command Description

C-8 Example 4 ─ Acquiring ASCII Data, Arbitrary Block

Example 4 – Code Listing

```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;

namespace Example4
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";

            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;

            string responseStringN = null;
            string responseStringL = null;
            string[] splitN = null;
            string[] splitL = null;

            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;

                //Timeout to 1 second (1000 ms)
                mbSession.Timeout = 1000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;
```

```
//We'll limit the number of data points to keep buffer small
mbSession.DefaultBufferSize = 5000;
testClear(mbSession);

//This next code group of code sets the sweep to 5 points,
//then makes trace1 active, sets to Smith Chart,
//does a trigger sweep and waits for sweep to finish.
//Then it sets output format to ASCII and asks for Final Data.
//Set the Language to NATIVE
mbSession.Write("LANG NATIVE\n");
mbSession.Write(":SENSE:SWEEP:POINTS 5\n");
mbSession.Write("CH1;SMI;\n");
mbSession.Write("TRS;WFS;HLD\n");
mbSession.Write("FMA;OFD\n");
responseStringN = testStatus_ReadArb(mbSession);
//the results are delimited by a newline (\n)
splitN = responseStringN.Split('\n');

//Set the Language to LIGHTNING
//Then we set the LANG LIGHT to see the subtle differences in
//how data is transferred when VectorStar is in Lightning mode.
mbSession.Write("LANG LIGHT\n");
mbSession.Write("FMA;OFD\n");
responseStringL = testStatus_ReadArb(mbSession);
//the results are delimited by a comma
splitL = responseStringL.Split(',');

//Send results to a file
StreamWriter output = new StreamWriter("OFD.txt");
output.WriteLine("Native Results");
foreach (string s in splitN)
    output.WriteLine(s);

output.WriteLine("Lightning Results");
foreach (string s in splitL)
    output.WriteLine(s);
output.Close();
mbSession.Write("RTL\n");
mbSession.Dispose();
```

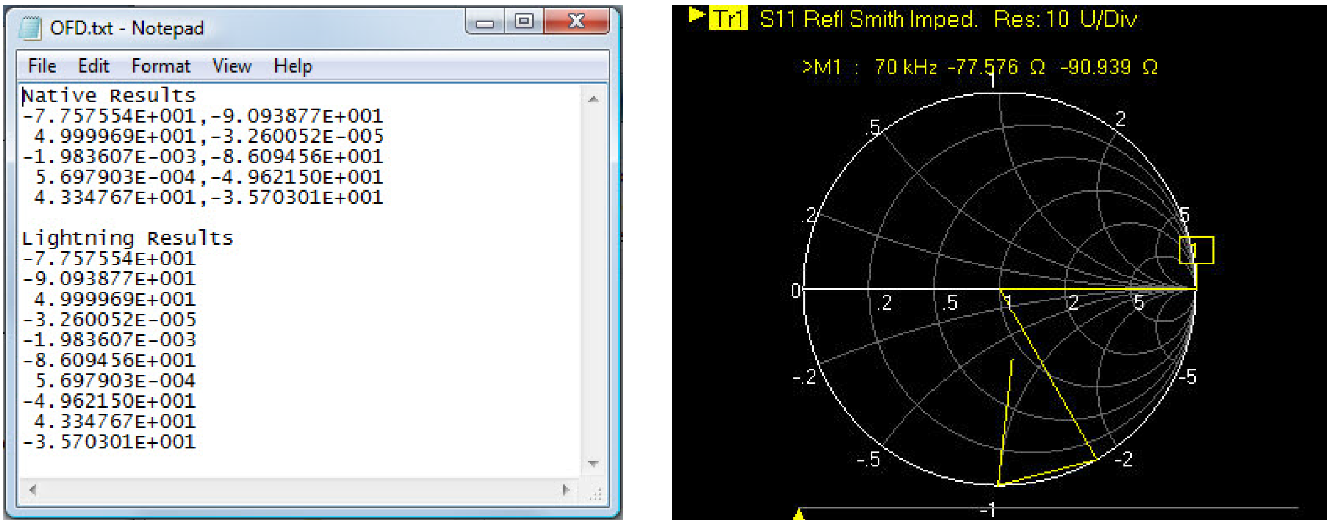
```
        }
        catch (VisaException v_exp)
        {
            Console.WriteLine("Visa caught an error!!");
            Console.WriteLine(v_exp.Message);
        }

        catch (Exception exp)
        {
            Console.WriteLine("Something didn't work!!");
            Console.WriteLine(exp.Message);
        }
    }

    private static void testClear(MessageBasedSession mbSession)
    {
        mbSession.Write("*CLS\n");
    }

    private static string testStatus_ReadArb(MessageBasedSession mbSession)
    {
        //These are the bits to check.
        int b2 = 4,     //Error Queue is not empty
            b4 = 16;    //MAV = Message Available
        string responseString = null;
        string replyString = null;
        string errorString = null;

        //Read the Status Byte of Service Request Status Register.
        StatusByteFlags sb = mbSession.ReadStatusByte();

        while (((int)sb & (b2 + b4)) == 0)
        {
            Thread.Sleep(10);
            sb = mbSession.ReadStatusByte();
        }
        if (((int)sb & b2) != 0)
        {
            errorString = mbSession.Query("OGE\n");
            Console.WriteLine("Error Queue: " + errorString);
```

```
            }
            else if (((int)sb & b4) != 0)
            {
                responseString = mbSession.ReadString();
                replyString = stripHeader(responseString);

            }
            mbSession.Write("*CLS\n");
            return replyString;
        }


        private static string stripHeader(string responseString)
        {
            int i = 0;
            string strReturn = null;
            if (responseString[i++] == '#')
            {
                //Header is ASCII, get 2nd byte and convert to int.
                StringBuilder sCount = new StringBuilder(responseString, i++, 1, 1);
                int count1 = int.Parse(sCount.ToString());
                //now read the bytecount string and convert to int
                StringBuilder sBytes = new
StringBuilder(responseString,i,count1,count1);
                int count2 = int.Parse(sBytes.ToString());
                //set the index of the start of the data
                i += count1;
                //return the string with the header stripped off
                strReturn = responseString.Substring(i);
            }
            return strReturn;
        }
    }
}
```

Example 4 – Discussion

1. Output result values are the same for Native mode and Lightning mode (since we put the unit in HOLD) but the delimiters are different.

- Native mode uses a comma to separate the two values at each frequency and a newline (\n) to separate pairs of values while Lightning mode separates each value with a comma. The display was set for Smith Chart, impedance.



OFD command output on left. Resultant Smith Chart on right.

Figure C-19. Example 4 – Listing and Graph Outputs

2. The OFD command returns an arbitrary block containing either ASCII or Binary data depending on the currently selected format.

- In this example we select ASCII using the FMA command.
- The Arbitrary block header is stripped off with the `stripHeader(string responseString)` function.
- The next example shows how to get binary data.

C-9 Example 5 – Acquiring Binary Data, Arbitrary Block

Example 5 – Code Listing

```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;

namespace Example5
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";

            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;

            double[] responseArray = null;

            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;

                //Timeout to 1 second (1000 ms)
                mbSession.Timeout = 1000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;

                //We'll limit the number of data points to keep buffer small
                mbSession.DefaultBufferSize = 5000;
```

```
            testClear(mbSession);

            //Set the Language to NATIVE
            //This code group sets the sweep to 8 points,
            //then makes trace2 active, sets to Mag-Phase, does a trigger sweep,
            //and waits for sweep to finish.
            //Then it sets output format to 64 bit binary and
            //least significant bit first (little endian) and
            //asks for Final Data.
            mbSession.Write("LANG NATIVE\n");
            mbSession.Write("CH2;MPH;\n");
            mbSession.Write(":SENSE:SWEEP:POINTS 8\n");
            mbSession.Write("TRS;WFS;HLD\n");
            mbSession.Write("LSB;FMB;OFD\n");
            responseArray = testStatus_ReadArbBinaryDouble(mbSession);
            //responseArray[] will hold the 16 readings from the VNA.
            //They will be mag1, phase1, mag2, phase2, etc.

            //Send results to a file
            StreamWriter output = new StreamWriter("OFD.txt");
            output.WriteLine("Native Results (Mag-Phase)");
            foreach (double d in responseArray)
                output.WriteLine(d);
            output.Close();
            mbSession.Write("RTL\n");
            mbSession.Dispose();
        }
        catch (VisaException v_exp)
        {
            Console.WriteLine("Visa caught an error!!");
            Console.WriteLine(v_exp.Message);
        }

        catch (Exception exp)
        {
            Console.WriteLine("Something didn't work!!");
            Console.WriteLine(exp.Message);
        }

    }
```

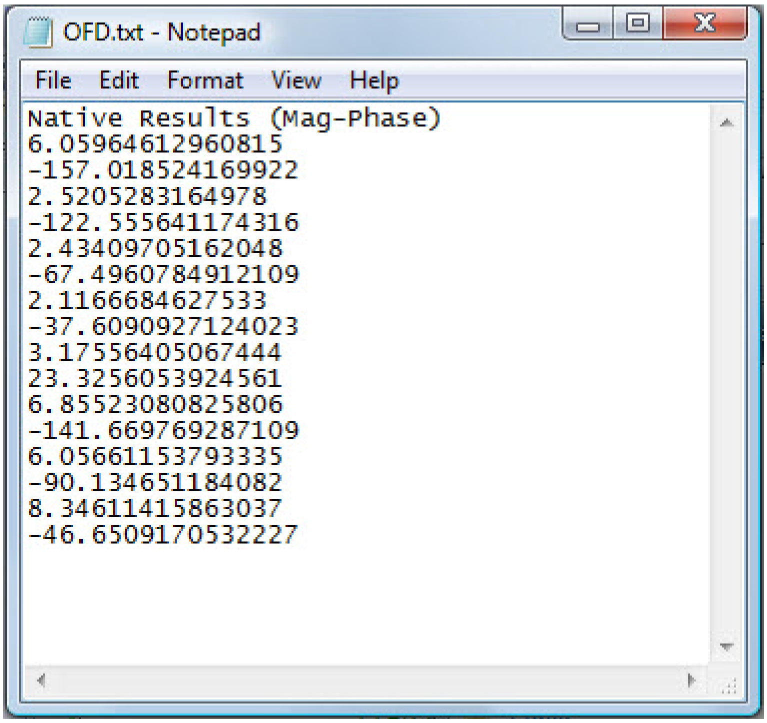
```
        private static double[] testStatus_ReadArbBinaryDouble(MessageBasedSession
mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            byte[] responsebytes = null;
            double[] replybytes = null;
            string errorString = null;

            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();

            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                errorString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + errorString);
            }
            else if (((int)sb & b4) != 0)
            {
                //Here we use the ReadByteArray() function to read the
                //binary data into a byte array.
                //Then we convert the bytes to doubles.
                responsebytes = mbSession.ReadByteArray();
                replybytes = arbToDouble(responsebytes);

            }
            mbSession.Write("*CLS\n");
            return replybytes;
        }

        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
```

```
        }

        //Here we convert Arb Block Binary Data to a double array
        //This function reads the arbitrary block header and then converts
        //the byte array into a double array. Every 8 bytes is converted to
        //a double. We use a MemoryStream and BinaryReader to do the conversion.
        private static double[] arbToDouble(byte[] responseBytes)
        {
            int i = 0;
            double[] dReturn = null;

            //Arbitrary Block should start with a #
            if (responseBytes[i++] == '#')
            {
                //Header is ASCII, get 2nd byte and convert to int
                string sCount = ASCIIEncoding.ASCII.GetString(responseBytes, i++, 1);
                int count1 = int.Parse(sCount);
                //now read the bytecount string and convert to int
                string sBytes = ASCIIEncoding.ASCII.GetString(responseBytes, i,
count1);
                int count2 = int.Parse(sBytes);
                //the number of doubles is the #bytes/sizeof(double)
                int dataCount = count2 / sizeof(double);
                //resize the response array
                dReturn = new double[dataCount];
                //set the index of the start of the data
                i += count1;

                //There are many ways to convert a byte array to a double array
                MemoryStream stream = new MemoryStream(responseBytes, i, count2);
                //BinaryReader reads this data type in little-endian format
                //So we must use the LSB mnemonic when acquiring the data
                BinaryReader reader = new BinaryReader(stream);
                for (int ii = 0; ii < dataCount; ii++)
                {
                    dReturn[ii] = reader.ReadDouble();
                }
             }
            return dReturn;
        }
```

```
    }
}
```

Example 5 – Discussion

Output file from this program should be in \Example5\bin\Debug\OFD.txt.



ODF command output listing.

Figure C-20. Example 5 – OFD Output

C-10 Example 6 – Output S2P File

Example 6 – Code Listing

```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;

namespace Example7
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            string responseStringN = null;
            string[] splitN = null;

            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;

                //Timeout to 3 second (1000 ms)
                mbSession.Timeout = 3000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;

                //We'll limit the number of data points to keep buffer small
                mbSession.DefaultBufferSize = 5000;
                testClear(mbSession);
```

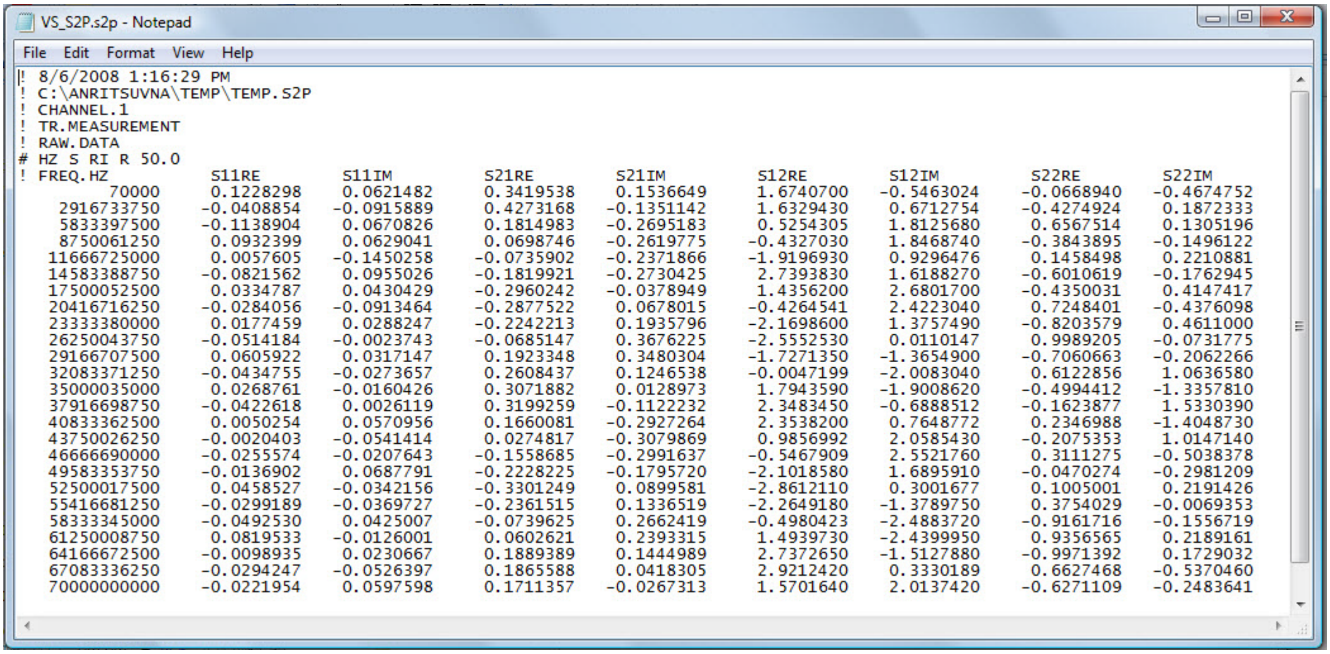
```
            //Set the Language to NATIVE
            mbSession.Write("LANG NATIVE\n");
            mbSession.Write(":SENSE:SWEEP:POINTS 25\n");
            //set format to Hz and Re/Im
            //We set the format of the S2P file parameters to
            //Frequency in Hz and data in Real/Imag format.
            mbSession.Write(":FORM:SNP:FREQ HZ\n");
            mbSession.Write(":FORM:SNP:PAR REIM\n");
            //Output S2P file over GPIB
            mbSession.Write("TRS;WFS;OS2P\n");
            responseStringN = testStatus_ReadArb(mbSession);
            //the results are delimited by a newline (\n)
            //Here the SplitN string array will hold each line of the S2P file.
            splitN = responseStringN.Split('\n');

            //Send results to a file
            StreamWriter output = new StreamWriter("VS_S2P.s2p");
            foreach (string s in splitN)
                output.WriteLine(s);
            output.Close();
            mbSession.Write("RTL\n");
            mbSession.Dispose();
        }
        catch (VisaException v_exp)
        {
            Console.WriteLine("Visa caught an error!!");
            Console.WriteLine(v_exp.Message);
        }

        catch (Exception exp)
        {
            Console.WriteLine("Something didn't work!!");
            Console.WriteLine(exp.Message);
        }
    }

    private static void testClear(MessageBasedSession mbSession)
    {
        mbSession.Write("*CLS\n");
```

```
        }

        private static string testStatus_ReadArb(MessageBasedSession mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            string responseString = null;
            string replyString = null;
            string errorString = null;

            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();

            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                errorString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + errorString);

            }
            else if (((int)sb & b4) != 0)
            {
                responseString = mbSession.ReadString();
                replyString = stripHeader(responseString);
            }
            mbSession.Write("*CLS\n");
            return replyString;
        }

        private static string stripHeader(string responseString)
        {
            int i = 0;
            string strReturn = null;
            if (responseString[i++] == '#')
            {
```

```
                //Header is ASCII, get 2nd byte and convert to int.
                StringBuilder sCount = new StringBuilder(responseString, i++, 1, 1);
                int count1 = int.Parse(sCount.ToString());
                //Now read the bytecount string and convert to int.
                StringBuilder sBytes = new StringBuilder(responseString, i, count1, 
count1);
                int count2 = int.Parse(sBytes.ToString());
                //Set the index of the start of the data.
                i += count1;
                //return the string with the header stripped off
                strReturn = responseString.Substring(i);
            }
            return strReturn;
        }
    }
}
```

Example 6 – Discussion

Expected output is in \Example6\Bin\Debug\VS_S2P.s2p.



VS_S2P.s2p - Notepad

File Edit Format View Help

```
! 8/6/2008 1:16:29 PM
! C:\ANRITSUVNA\TEMP\TEMP.S2P
! CHANNEL.1
! TR.MEASUREMENT
! RAW.DATA
# HZ S RI R 50.0
! FREQ.HZ          S11RE        S11IM         S21RE        S21IM         S12RE        S12IM         S22RE        S22IM
          70000    0.1228298    0.0621482     0.3419538    0.1536649     1.6740700   -0.5463024    -0.0668940   -0.4674752
     2916733750   -0.0408854   -0.0915889     0.4273168   -0.1351142     1.6329430    0.6712754    -0.4274924    0.1872333
     5833397500   -0.1138904    0.0670826     0.1814983   -0.2695183     0.5254305    1.8125680     0.6567514    0.1305196
     8750061250    0.0932399    0.0629041     0.0698746   -0.2619775    -0.4327030    1.8468740    -0.3843895   -0.1496122
    11666725000    0.0057605   -0.1450258    -0.0735902   -0.2371866    -1.9196930    0.9296476     0.1458498    0.2210881
    14583388750   -0.0821562    0.0955026    -0.1819921   -0.2730425     2.7393830    1.6188270    -0.6010619   -0.1762945
    17500052500    0.0334787    0.0430429    -0.2960242   -0.0378949     1.4356200    2.6801700    -0.4350031    0.4147417
    20416716250   -0.0284056   -0.0913464    -0.2877522    0.0678015    -0.4264541    2.4223040     0.7248401   -0.4376098
    23333380000    0.0177459    0.0288247    -0.2242213    0.1935796    -2.1698600    1.3757490    -0.8203579    0.4611000
    26250043750   -0.0514184   -0.0023743    -0.0685147    0.3676225    -2.5552530    0.0110147     0.9989205   -0.0731775
    29166707500    0.0605922    0.0317147     0.1923348    0.3480304    -1.7271350   -1.3654900    -0.7060663   -0.2062266
    32083371250   -0.0434755   -0.0273657     0.2608437    0.1246538    -0.0047199   -2.0083040     0.6122856    1.0636580
    35000035000    0.0268761   -0.0160426     0.3071882    0.0128973     1.7943590   -1.9008620    -0.4994412   -1.3357810
    37916698750   -0.0422618    0.0026119     0.3199259   -0.1122232     2.3483450   -0.6888512    -0.1623877    1.5330390
    40833362500    0.0050254    0.0570956     0.1660081   -0.2927264     2.3538200    0.7648772     0.2346988   -1.4048730
    43750026250   -0.0020403   -0.0541414     0.0274817   -0.3079869     0.9856992    2.0585430    -0.2075353    1.0147140
    46666690000   -0.0255574   -0.0207643    -0.1558685   -0.2991637    -0.5467909    2.5521760     0.3111275   -0.5038378
    49583353750   -0.0136902    0.0687791    -0.2228225   -0.1795720    -2.1018580    1.6895910    -0.0470274   -0.2981209
    52500017500    0.0458527   -0.0342156    -0.3301249    0.0899581    -2.8612110    0.3001677     0.1005001    0.2191426
    55416681250   -0.0299189   -0.0369727    -0.2361515    0.1336519    -2.2649180   -1.3789750     0.3754029   -0.0069353
    58333345000   -0.0492530    0.0425007    -0.0739625    0.2662419    -0.4980423   -2.4883720    -0.9161716   -0.1556719
    61250008750    0.0819533   -0.0126001     0.0602621    0.2393315     1.4939730   -2.4399950     0.9356565    0.2189161
    64166672500   -0.0098935    0.0230667     0.1889389    0.1444989     2.7372650   -1.5127880    -0.9971392    0.1729032
    67083336250   -0.0294247   -0.0526397     0.1865588    0.0418305     2.9212420    0.3330189     0.6627468   -0.5370460
    70000000000   -0.0221954    0.0597598     0.1711357   -0.0267313     1.5701640    2.0137420    -0.6271109   -0.2483641
```

Transfer of an S2P file to the PC.

Figure C-21. Example 6 – S2P File to PC

C-11 Example 7 – Output a Bitmap

Example 7 – Code Listing

```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;

namespace Example6
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";

            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;

            byte[] responseArray = null;

            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;

                //Timeout to 3 seconds (3000 ms)
                mbSession.Timeout = 3000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;

                //We expect about 534k bytes
                //BMP files can be rather large so set up the buffer
```

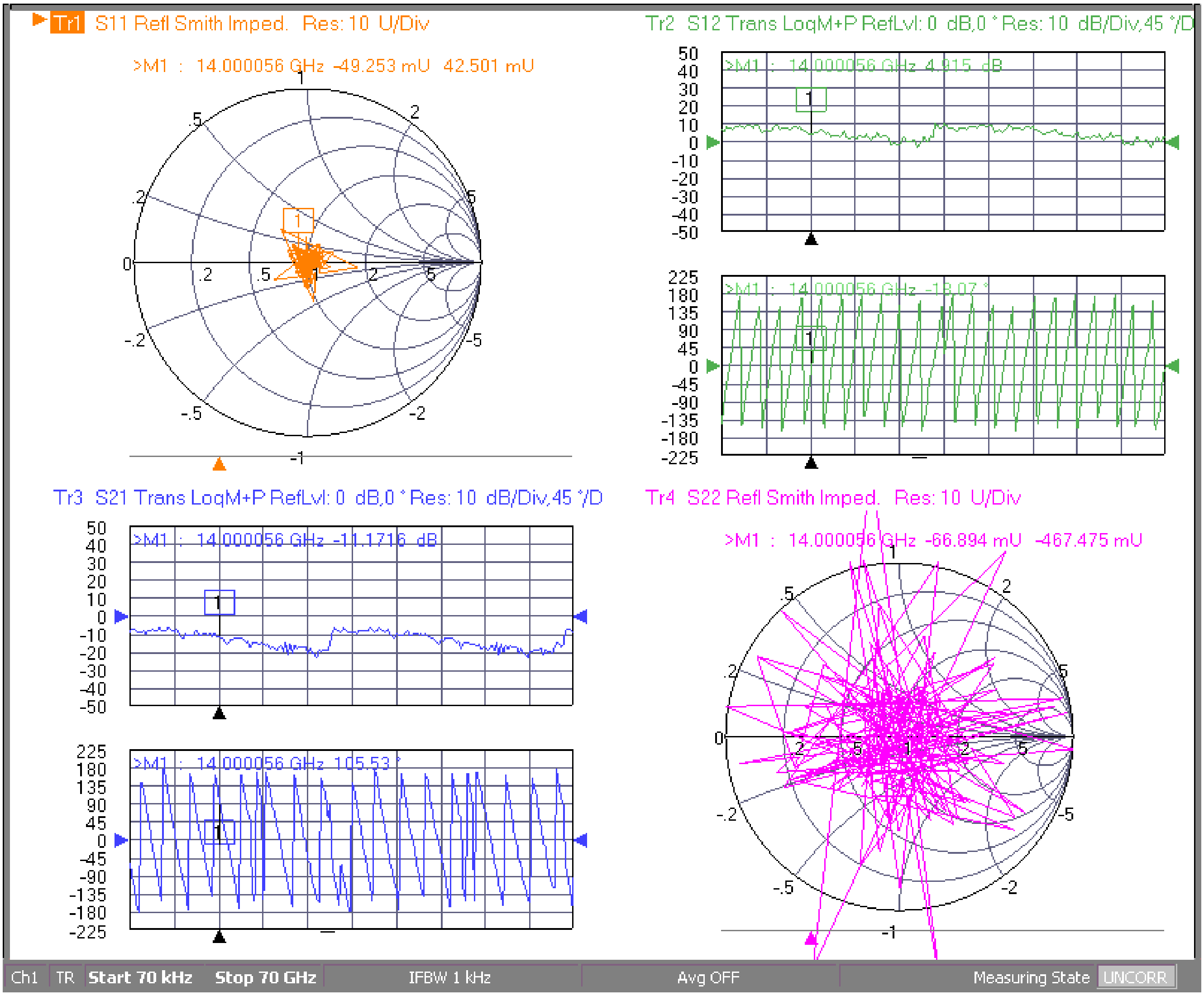
```
                //size to read it all at once.
                mbSession.DefaultBufferSize = 600000;
                testClear(mbSession);

                //Set the Language to LIGHT,
                mbSession.Write("LANG LIGHT\n");
                //set color on white and output bitmap over gpib
                mbSession.Write("BMPC;OBMP\n");
                responseArray = testStatus_ReadArbBinaryByte(mbSession);

                //Send results to a new Binary file
                BinaryWriter output = new BinaryWriter(File.Open("VS.bmp",
FileMode.Create));
                output.Write(responseArray);
                output.Close();
                mbSession.Write("RTL\n");
                mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }

            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
            }

        }

        private static byte[] testStatus_ReadArbBinaryByte(MessageBasedSession
mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            byte[] responsebytes = null;
            byte[] replybytes = null;
```

```
        string errorString = null;

        //Read the Status Byte of Service Request Status Register
        StatusByteFlags sb = mbSession.ReadStatusByte();

        while (((int)sb & (b2 + b4)) == 0)
        {
            Thread.Sleep(10);
            sb = mbSession.ReadStatusByte();
        }
        if (((int)sb & b2) != 0)
        {
            errorString = mbSession.Query("OGE\n");
            Console.WriteLine("Error Queue: " + errorString);

        }
        else if (((int)sb & b4) != 0)
        {
            //Here we use the ReadByteArray() function to read the
            //binary data into a byte array. But we still want to
            //strip off the arb block header.
            responsebytes = mbSession.ReadByteArray();
            replybytes = arbToByte(responsebytes);

        }
        mbSession.Write("*CLS\n");
        return replybytes;
    }

    private static void testClear(MessageBasedSession mbSession)
    {
        mbSession.Write("*CLS\n");
    }

    //Convert Arb Block Binary Data to Byte array
    //This function reads the arbitrary block header and then grabs and
    //returns the byte array data.
    //We continue to use the MemoryStream and BinaryReader classes.
    private static byte[] arbToByte(byte[] responseBytes)
    {
```

```
            int i = 0;
            byte[] dReturn = null;

            //Arbitrary Block should start with a #
            if (responseBytes[i++] == '#')
            {
                //Header is ASCII, get 2nd byte and convert to int
                string sCount = ASCIIEncoding.ASCII.GetString(responseBytes, i++, 1);
                int count1 = int.Parse(sCount);
                //now read the bytecount string and convert to int
                string sBytes = ASCIIEncoding.ASCII.GetString(responseBytes, i,
count1);
                int count2 = int.Parse(sBytes);
                //the number of doubles is the #bytes/sizeof(double)
                int dataCount = count2 / sizeof(byte);
                //resize the response array
                dReturn = new byte[dataCount];
                //set the index of the start of the data
                i += count1;

                MemoryStream stream = new MemoryStream(responseBytes, i, count2);
                //BinaryReader reads this data type in little-endian format
                //So we must use the LSB mnemonic when acquiring the data
                BinaryReader reader = new BinaryReader(stream);
                for (int ii = 0; ii < dataCount; ii++)
                {
                    dReturn[ii] = reader.ReadByte();
                }
            }
            return dReturn;
        }
    }
}
```

Example 7 – Discussion

Output should be in \Example6\Bin\Debug\VS.bmp.



Results from Example 6.

Figure C-22. Example 6 – Output Results as BMP File

Property	Value
Image	
Dimensions	804 x 663
Width	804 pixels
Height	663 pixels
Bit depth	8

Properties of the bitmap file in this example.

Figure C-23. Bitmap File Properties

Appendix D — Programming Basics with Legacy Software

D-1 Introduction

This appendix provides an overview of programming techniques for controlling the VectorStar Series VNAs with older legacy software. Best practices recommend using the techniques and software described in Appendix A, B, and C for new development. This appendix describes using the NI-VISA driver and C# over GPIB, TCP/IP (using VXI-11), and USB. Programming using TCP/IP Sockets requires some special handling and is not discussed in this document.

D-2 Programming Basics

Programmatic control of the VNA allow automating a test sequence, orchestrating a complex measurement involving various pieces of test equipment, gathering a time series of data, or a convenient way of getting data or images transferred from the VNA to a PC for further analysis.

VISA

VISA (Virtual Instrument System Architecture) is an I/O software standard for communicating with test instruments like VectorStar over any of the bus architectures which VectorStar supports. Licensed VISA drivers are available from National Instruments and Agilent. National Instruments VISA drivers are available for the following operating systems:

- Windows 2000/Vista x64/Vista x86/XP/Windows 7
- Mac OS X
- Linux
- Mandriva
- RedHat
- SUSE
- PharLap
- VxWorks
- Pocket PC 2003
- Windows CE/Mobile

Note	The NI-VISA software, a product of National Instruments, is not freeware, but rather a licensed software product available for purchase from National Instruments. The type of license required depends whether you are developing or deploying an application. More information on National Instruments and NI-VISA, consult their web site at http://www.ni.com.

It's always a good idea to get the latest driver (Version 4.4.1 as of the release date of this document), but make sure to get the Full Version (not just the runtime) version 4.4 or higher for the best support of the latest .NET 3.5 Framework, USB and TCP/IP. The driver is available from http://www.ni.com or from NI Device Driver CD that comes with NI hardware.

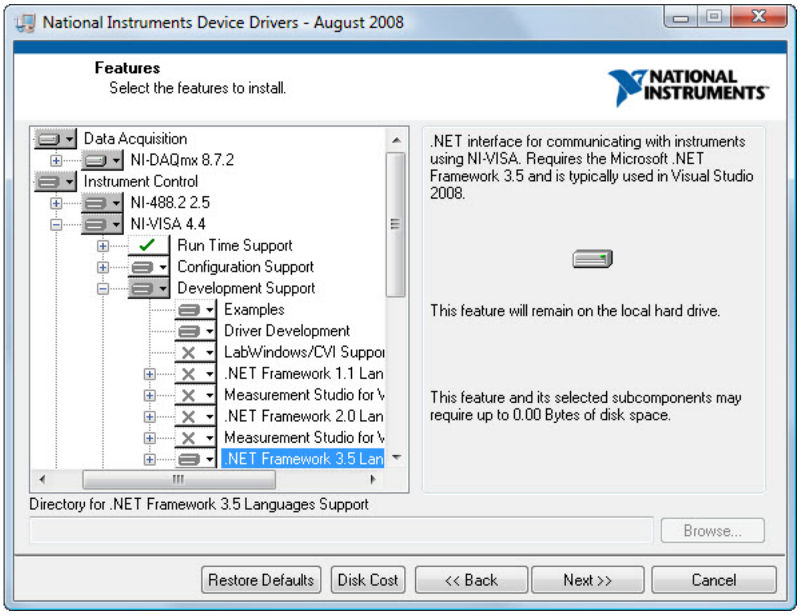


Figure D-1. Installing NI-VISA 4.4 Driver from the Device Driver CD

Installing NI-VISA 4.4 Driver from the Device Driver CD does not automatically install Development Support for the .NET Framework so it must be selectively installed. Be sure to select “Examples” as well. The VISA Version 4.4 driver supports the .NET Framework 1.1, 2.0 and 3.5.

The Full NI-VISA 4.4.1 installation can also be selected from Developer Support for .NET Framework 3.5.

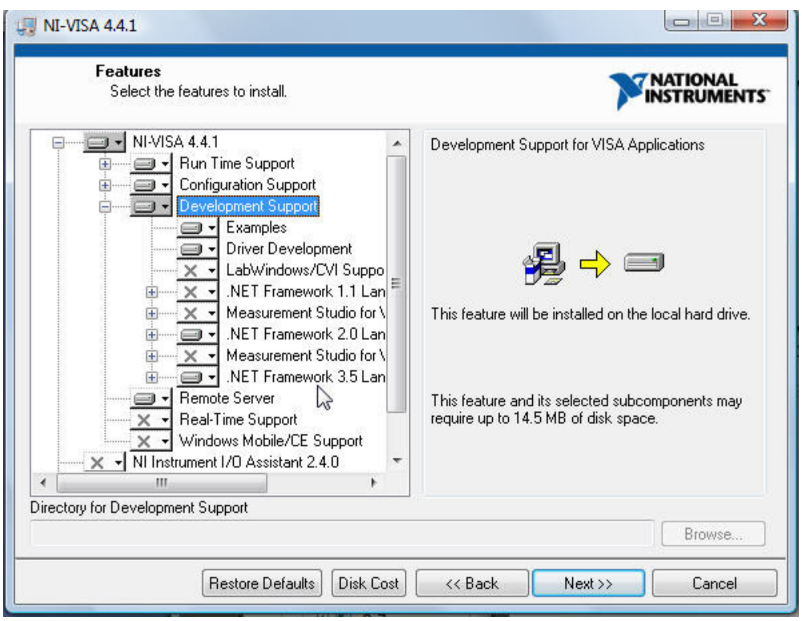


Figure D-2. Full NI-VISA 4.4.1 Installation

Search for “VisaNS” in Visual Studio (Local) Help to find more information about using the VISA driver.

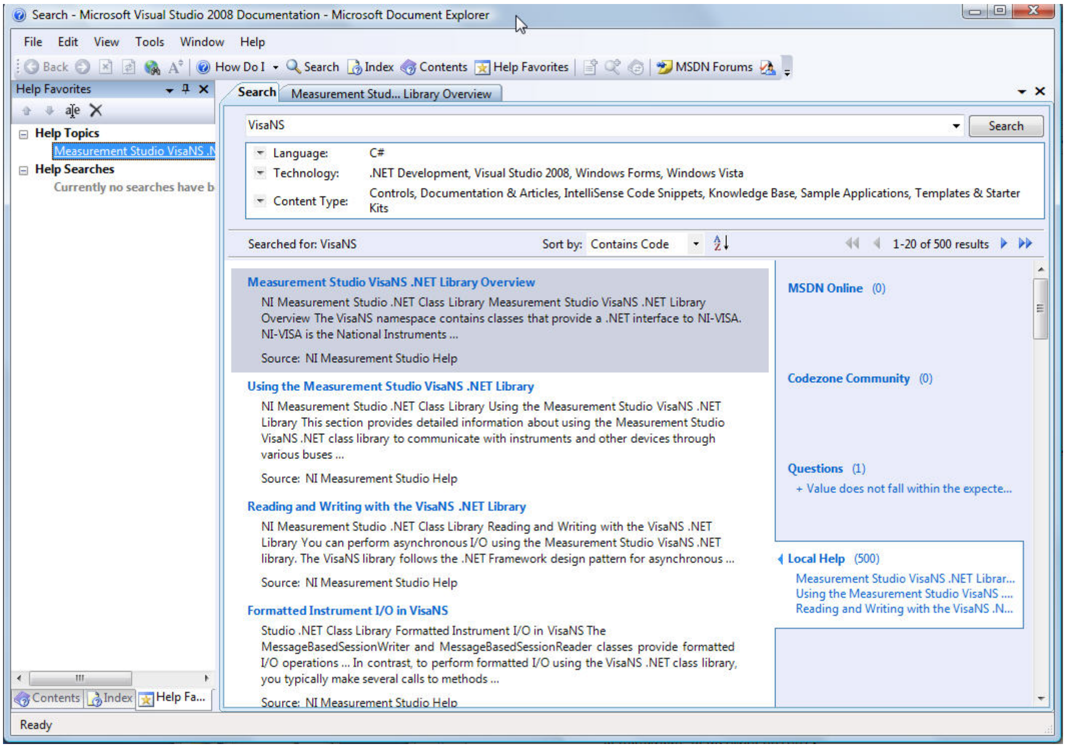


Figure D-3. Search for “VisaNS”

Programming Environments

Programming environments that are commonly used with test equipment include LabVIEW, LabWindows/CVI, Microsoft Visual Studio (2003, 2005, 2008), Visual Basic 6, HP Basic, and others. NI-VISA supports the following Windows development environments:

- LabVIEW 8.6/8.5/8.2/7.1.1
- LabWindows/CVI 8.5/8.1/8.0/7.1
- Measurement Studio for Visual Studio 2008/2005/2003 (all versions)
- Microsoft Visual Studio 2008/2005/2003/6.0
- Microsoft Visual Basic 6.0

The examples below use Visual C# 2008 Express Edition, which uses C# 3.0 and the .NET Framework 3.5. You can download the Express Edition for free from the Microsoft at:
http://www.microsoft.com/express/download

GPIB Mnemonics

The MS464xB Series VNA has a fairly large set of GPIB commands. This includes a set of Native commands, Lightning commands, and for MS464xA Series VNAs, Agilent HP8510 commands. You use the same commands regardless of the communication method employed. Refer to Example 3 in this appendix for using the LIST command to output the full set of supported GPIB mnemonics directly from the VNA.

D-3 Installing the LabVIEW Driver

The LabVIEW driver was developed and is supported by National Instruments. The driver is available at the National Instrument's Instrument Driver Network (http://www.ni.com/devzone/idnet/) or you can download it directly from within LabVIEW as shown below:

1. Install the instrument drivers from the LabVIEW help menu.

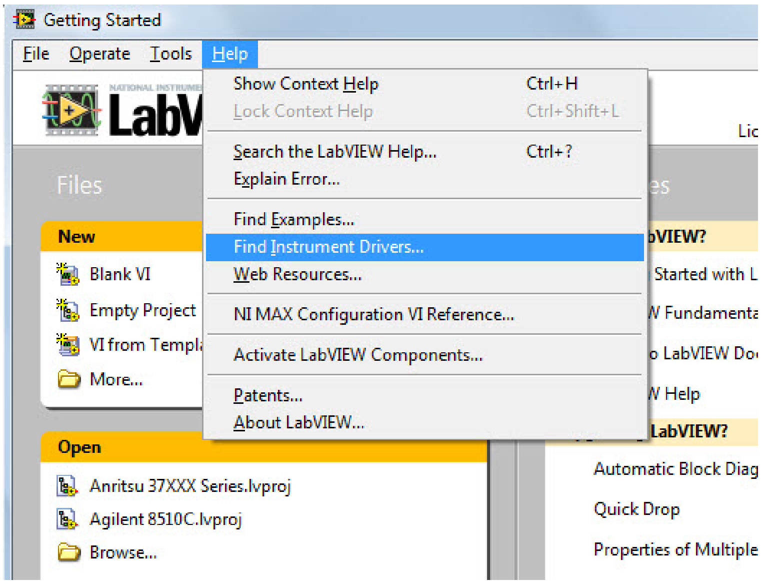


Figure D-4. LabVIEW Help Menu

2. In the Manufacturer field, search for "Anritsu." In the Additional Keywords field, use "37XXX."

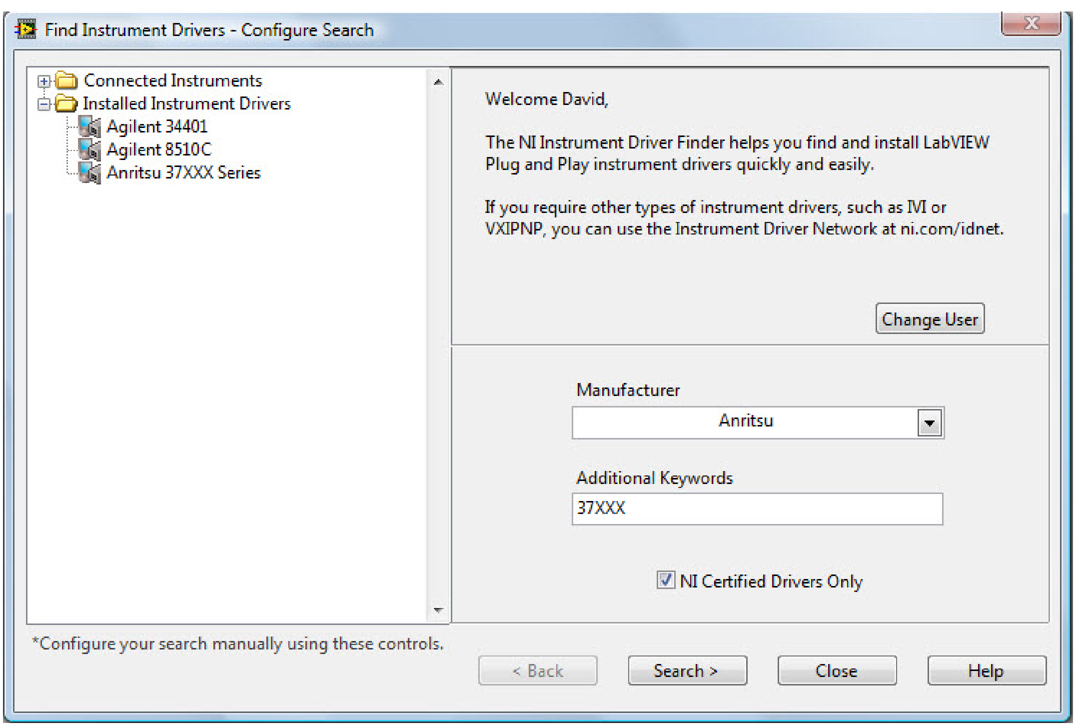


Figure D-5. LabVIEW Find Instrument Drivers - Configure Search Dialog

3. Select Install > to start the installation of the 37XXX driver for LabVIEW.

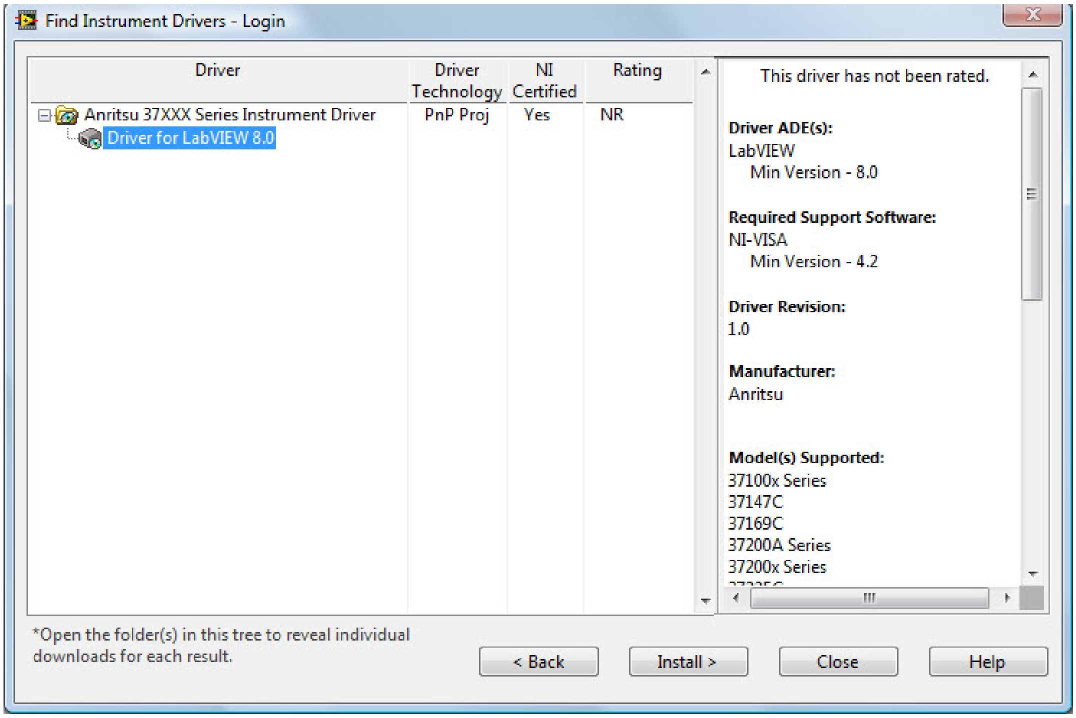


Figure D-6. LabVIEW Find Instrument Drivers - Login Dialog

4. Note the driver installation path.
   - A typical installation path would be:
     `C:\Program Files\National Instruments\LabVIEW8.6\instr.lib\Anritsu 37XXX Series`

5. In the MS464xB Series VNA application, select the “Lightning” remote language.

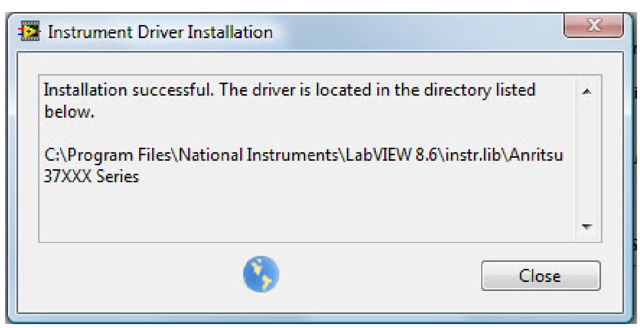
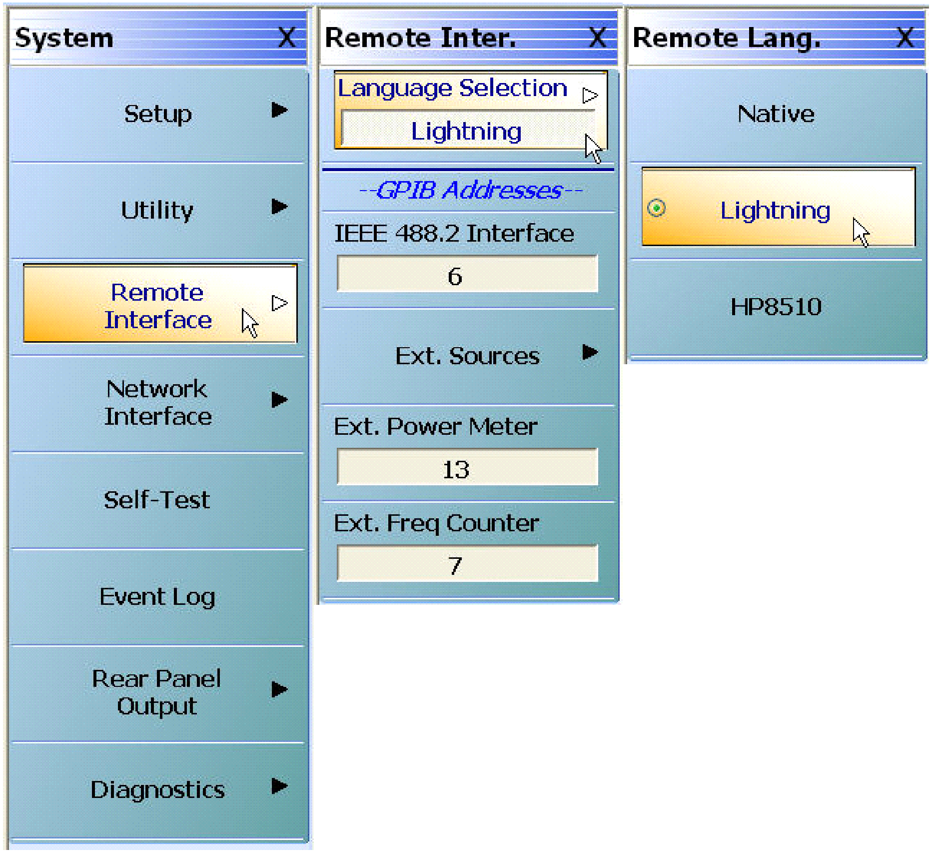


Figure D-7. LabVIEW Instrument Driver Installation Dialog



MAIN | System | SYSTEM | Remote Interface | REMOTE INTER. | Language Selection | REMOTE LANG.

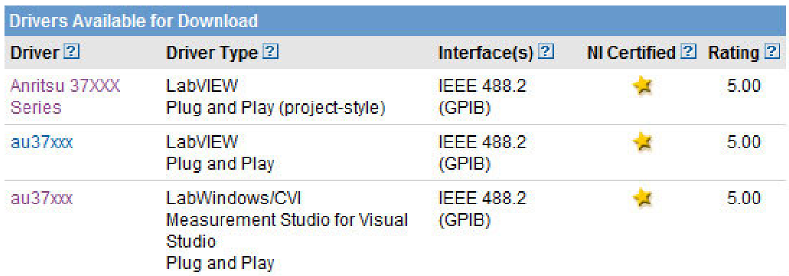
Figure D-8. REMOTE LANG. Menu – Remote Language Selection

Note	The remote language can also be set in the LabVIEW program, which is illustrated in the LabVIEW examples 3, 6 and 7.

D-4 Installing the LabWindows/CVI Driver

The LabWindows/CVI driver is developed and supported by National Instruments. The driver is available at the National Instrument's Instrument Driver Network (http://www.ni.com/devzone/idnet/).

1. The MS464xB Series VNA uses the au37xxx for LabWindows/CVI. Search for these drivers on the National Instruments Driver Network.

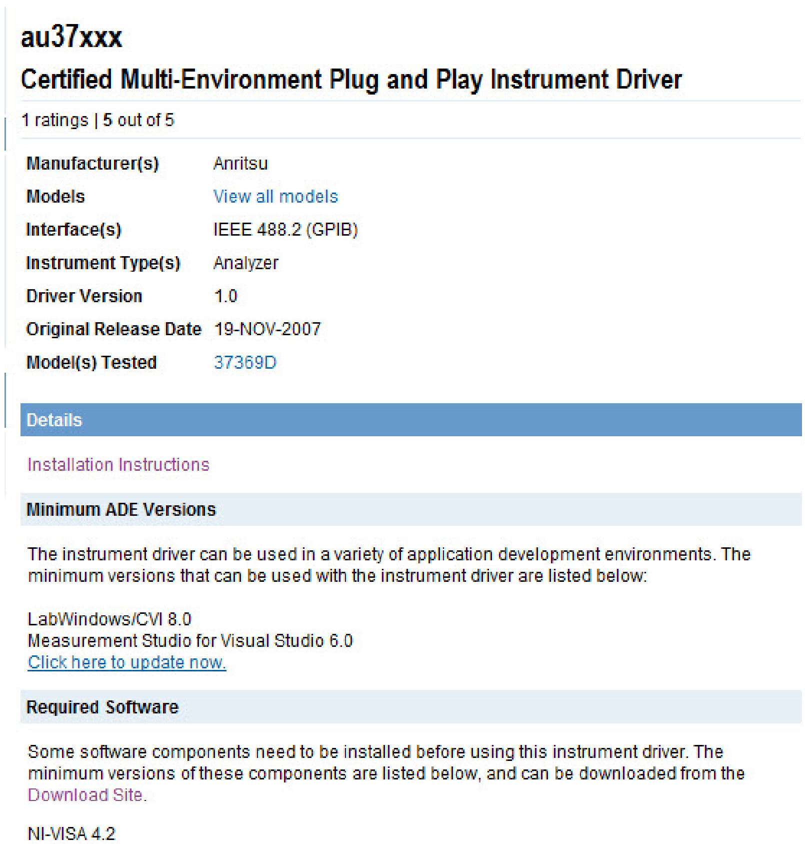


Drivers Available for Download

Driver	Driver Type	Interface(s)	NI Certified	Rating
Anritsu 37XXX Series	LabVIEW Plug and Play (project-style)	IEEE 488.2 (GPIB)		5.00
au37xxx	LabVIEW Plug and Play	IEEE 488.2 (GPIB)		5.00
au37xxx	LabWindows/CVI Measurement Studio for Visual Studio Plug and Play	IEEE 488.2 (GPIB)		5.00

Figure D-9. au37xxx LabWindows/CVI Drivers for the MS464xB Series VNA (http://www.ni.com/devzone/idnet/)

2. Click the au37xxx driver for LabWindows/CVI link for more details. Downloading the driver copies a zip file to your hard drive.



au37xxx

Certified Multi-Environment Plug and Play Instrument Driver

1 ratings | 5 out of 5

Manufacturer(s)	Anritsu
Models	View all models
Interface(s)	IEEE 488.2 (GPIB)
Instrument Type(s)	Analyzer
Driver Version	1.0
Original Release Date	19-NOV-2007
Model(s) Tested	37369D

Details

Installation Instructions

Minimum ADE Versions

The instrument driver can be used in a variety of application development environments. The minimum versions that can be used with the instrument driver are listed below:

LabWindows/CVI 8.0
Measurement Studio for Visual Studio 6.0
Click here to update now.

Required Software

Some software components need to be installed before using this instrument driver. The minimum versions of these components are listed below, and can be downloaded from the Download Site.

NI-VISA 4.2

Figure D-10. au37xxx LabWindows/CVI Driver Details

3. Click each Function Index of the driver to view a detailed HTML help file.

Hierarchical Function Index

- au37xxx_init
- Configure
  - Measurement Configuration
    - au37xxx_configureChannel
    - au37xxx_configureDomain
    - au37xxx_configureIFBandwidth
    - au37xxx_configureTrigger
    - au37xxx_configureAveraging
    - au37xxx_configureSmoothing
    - au37xxx_configureSamplers
    - au37xxx_configureMeasurementDelay
    - au37xxx_configureSpurReduction
    - au37xxx_configureSourceLock
    - au37xxx_configureFastCWMode
    - au37xxx_configureTestSignal
    - au37xxx_configureBiasHoldConditions
    - au37xxx_configureIFCalibration
    - au37xxx_configurePhaseLock
    - au37xxx_configureDUTIdentity
    - Sweep Configuration
      - au37xxx_configureSweep
      - au37xxx_configureSweepStimulus
      - au37xxx_configureSweepCW
      - au37xxx_configureDiscreteFrequencies
      - au37xxx_configureDiscreteFrequencyFill
    - Gate Configuration
      - au37xxx_configureGate
      - au37xxx_configureGateRange
    - Time Domain
      - au37xxx_configureTimeDomain
      - au37xxx_configureTimeDomainZoom
      - au37xxx_configureTimeDomainLowPass

Figure D-11. au37xxx LabWindows/CVI Driver Hierarchical Function Index

au37xxx_init

```
ViStatus au37xxx_init (ViRsrc resourceName, ViBoolean IDQuery, ViBoolean resetDevice, ViPSession instrumentHandle);
```

Purpose

This function performs the following initialization actions:
- Opens a session to the Default Resource Manager resource and a session to the specified device using the interface and address specified in the Resource_Name control.
- Performs an identification query on the Instrument.
- Resets the instrument to a known state.
- Sends initialization commands to the instrument that set any necessary programmatic variables such as Headers Off, Short Command form, and Data Transfer Binary to the state necessary for the operation of the instrument driver.
- Returns an Instrument Handle which is used to differentiate between different sessions of this instrument driver.
- Each time this function is invoked a Unique Session is opened. It is possible to have more than one session open for the same resource.

Parameters

Input

Name	Type	Description
resourceName	ViRsrc	This control specifies the interface and address of the device that is to be initialized (Instrument Descriptor). The exact grammar to be used in this control is shown in the note below. Default Value: "GPIB0::6::INSTR" Notes: (1) Based on the Instrument Descriptor, this operation establishes a communication session with a device. The grammar for the Instrument Descriptor is shown below. Optional parameters are shown in square brackets ([]). Interface Grammar -- GPIB GPIB[board]::primary address[::secondary address] [::INSTR] LAN TCPIP::tcp/ip address::socket_number::SOCKET The GPIB keyword is used for GPIB interface. The LAN keyword is used for LAN interface. Examples: (1) GPIB - "GPIB::20" (2) LAN - "TCPIP::192.168.1.33::5025::SOCKET" The default value for optional parameters are shown below. Optional Parameter Default Value --- board 0 secondary address none - 31

Figure D-12. au37xxx LabWindows/CVI Driver Function Help File

4. In the MS464xB Series VNA application, select the “Lightning” remote language.

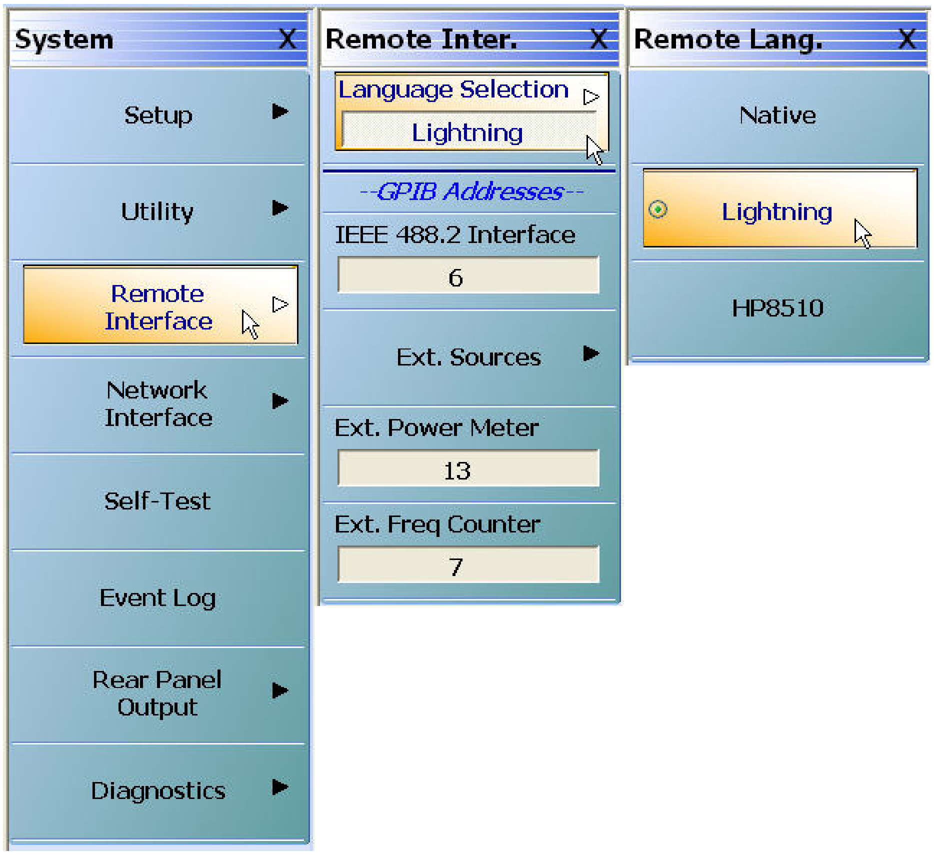


Figure D-13. SYSTEM, REMOTE INTER (Remote Interface) and REMOTE LANG (Remote Language) Menus

Note	The remote language can also be set in the LabWindows/CVI program, which is illustrated in the LabWindows/CVI examples 3, 6 and 7.

D-5 Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser

The Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser is a useful windows program. For more information about availability, consult the Anritsu web site at http://www.anritsu.com. The Help system of the exerciser provides an overview of the application.

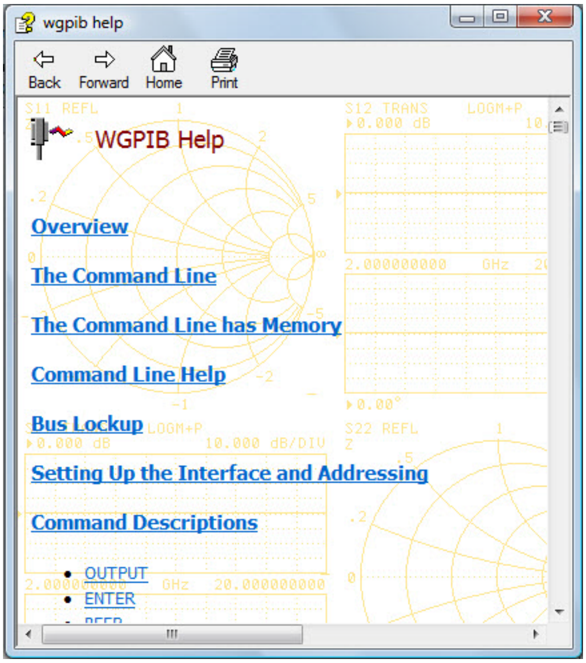


Figure D-14. Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser Help

Before writing any software, make sure that you can write to and read from the MS464xB Series VNA over the communication protocol you're interested in. The following dialog is an example of setting up a VXI-11 connection to VectorStar. The VectorStar TCP/IP address is 10.0.1.194 in this example.

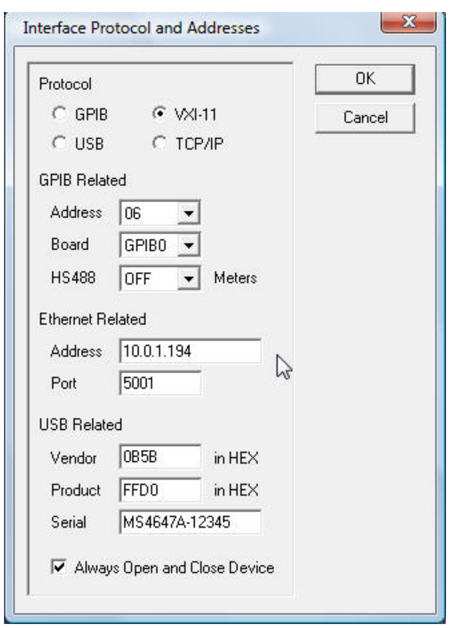


Figure D-15. Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser Interface Protocol and Address Dialog

The following shows the result of using the Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser to query the instrument using the "*IDN?" command.

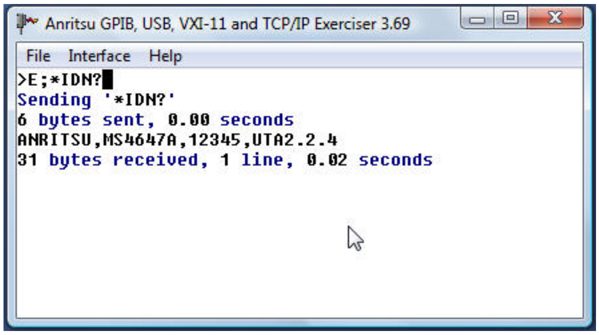


Figure D-16. Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser

Be sure to consult the following references for more detailed information on general programming:

- NI-VISA User Manual
- NI-VISA Programmer Reference Manual
- LabVIEW Fundamentals Manual
- Getting Started with LabWindows/CVI

D-6 VISA and C# Programming Examples

This section includes a few simple Console-based programs to demonstrate the use of the NI-VISA driver for controlling the MS464xB Series VNA.

Create a New Console Application Project

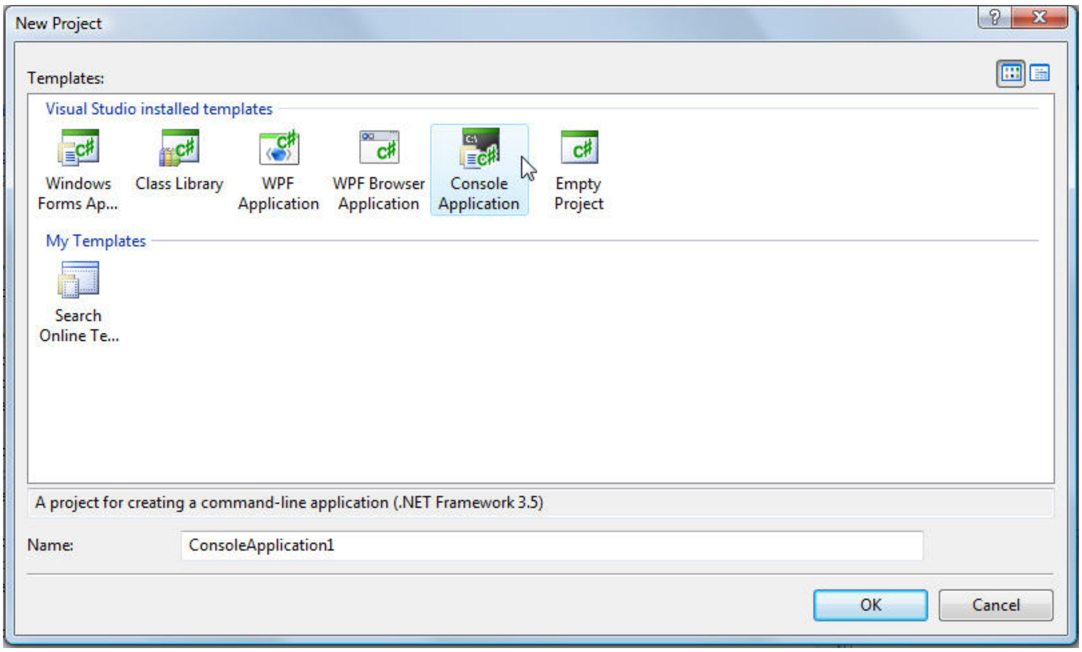


Figure D-17. New Project in Visual Studio

Add references for National Instruments Common and National Instruments VisaNS. The following shows the .NET 3.5 versions of these DLLs.

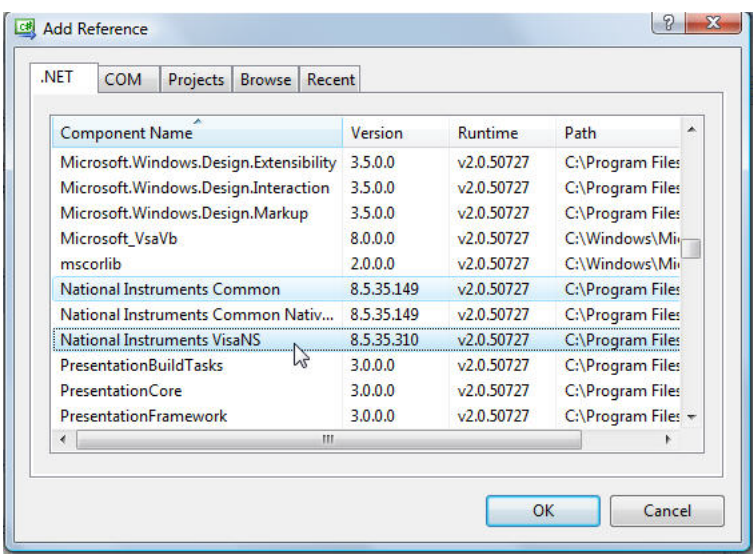


Figure D-18. Adding a Reference

For simple programs (like all the examples below), only these three references are required.

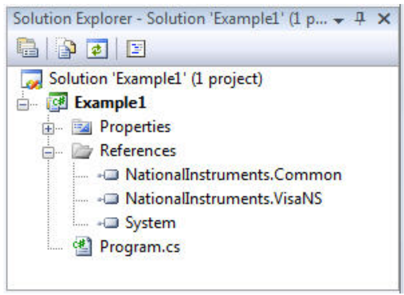


Figure D-19. Solution Explorer

If the references are set up correctly, copy and paste the code into the Program.cs file.

Build the Solution

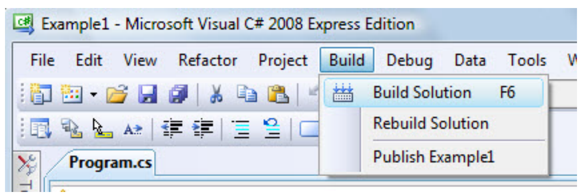


Figure D-20. Build Solution

Run the Code (Start Debugging).

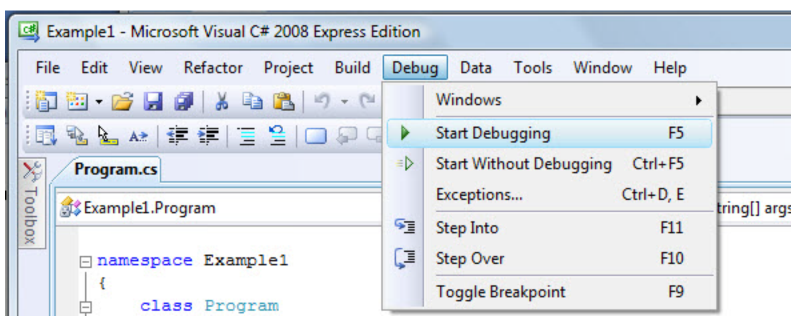


Figure D-21. Example Debug Menu

Each of the following examples include the complete program code.

Example 1 – Opening a Session and Sending the *IDN? Command

```
using System;
using NationalInstruments.VisaNS;
namespace Example1
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            //response string
            string responseString = null;
            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Send "*IDN?" command
                mbSession.Write("*IDN?\n");
                //Read the response
                responseString = mbSession.ReadString();
                //Write to Console
                Console.WriteLine("Response to *IDN?:");
                Console.WriteLine(responseString);
                //Return to Local Control
                mbSession.Write("RTL\n");
                //Close the Session
                mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
                Console.WriteLine();
            }
            keepConsoleUp();
        }
        private static void keepConsoleUp()
        {
            Console.WriteLine("");
            Console.WriteLine("Enter to Continue");
            Console.ReadLine();
        }
    }
}
```

Example 1 Discussion

Program Output

1. Examine the program output.
2. The console output should look something like the following if everything worked. The result is Manufacturer, Model, Serial Number, and Firmware Version.

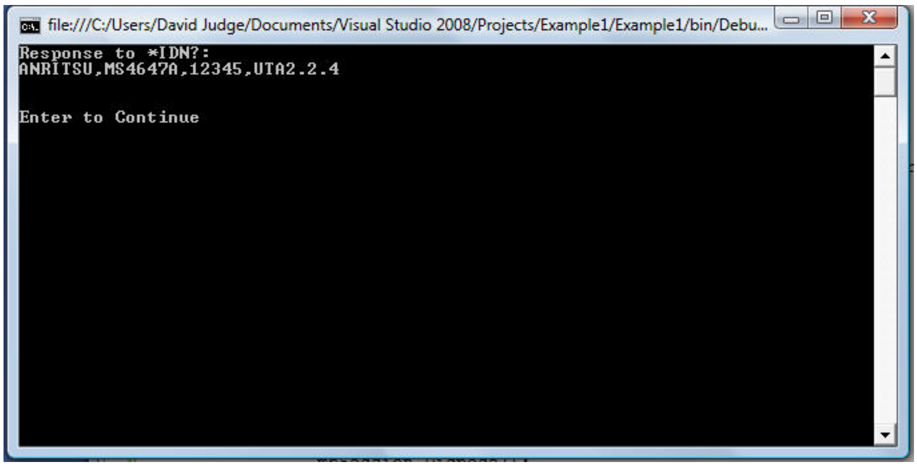


Figure D-22. Program Output

Address String

3. Check the Address String.
4. If the example did not work, check that the Address string matches your setup. The following error results if the address is wrong or the connection string is no good:

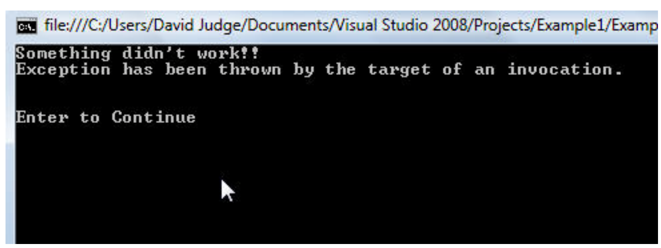


Figure D-23. Error Result

5. The first thing to observe is the VISA connection string. Here are some possible strings:

```
//VXI-11 Connection string
string sAddress = "TCPIP0::10.0.1.194::INSTR";
//GPIB Connection string
string sAddress = "GPIB0::6::INSTR";
//USB Connection string (vendor::product::serial_number)
string sAddress = "USB0::0x0B5B::0xFFD0::MS4647B-12345::INSTR";
```

6. When using VISA, the only thing that needs to be changed for any of these possible communication protocols is the connection string. The rest of the code should be exactly the same (except for SOCKETS). VXI-11 should be used for TCP/IP since it implements the IEEE 488.2 standard and all status checking.

7. The connection string for a TCP/IP socket connection is:

```
//TCP/IP Sockets Connection string
string sAddress = "TCPIP0::10.0.1.194::5001::SOCKET";
```

8. Opening the communication session:

```
//open a Session to the VNA
mySession = ResourceManager.GetLocalManager().Open(sAddress);
//cast this to a message based session
mbSession = (MessageBasedSession)mySession;
```

9. Write to VNA and Read from VNA:

```
//Send "*IDN?" command
mbSession.Write("*IDN?\n");
//Read the response
responseString = mbSession.ReadString();
```

Notice that the `mbSession.Write()` commands are terminated with a “/n”. This is the newline character.

Session Parameters

10. The next example discusses session parameters. Some of the other examples will get into reading binary data and arbitrary block data.

Exception Handling

11. The VISA driver catches certain types of errors like time-outs and .NET catches the rest.

```
catch (VisaException v_exp)
{
    Console.WriteLine("Visa caught an error!!");
    Console.WriteLine(v_exp.Message);
}
catch (Exception exp)
{
    Console.WriteLine("Something didn't work!!");
    Console.WriteLine(exp.Message);
    Console.WriteLine();
}
```

The `VisaException` function catches a time-out on a `ReadString()`.

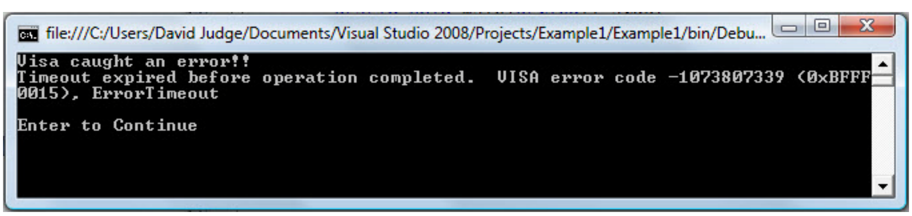


Figure D-24. Error Catching

Close the Session

12. Close the session using `Dispose()`:

```
//Close the Session
mbSession.Dispose();
```

Example 2 – Session Parameters and Status Checking

```
using System;
using NationalInstruments.VisaNS;
using System.Threading;
namespace Example2
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Now set some Session Parameters
                //Timeout to 1 second (1000 ms)
                mbSession.Timeout = 1000;
                //use newline (\n or 0x0a) as termination character
                mbSession.TerminationCharacter = 0x0a;
                //terminate reads if the termination character is spotted
                mbSession.TerminationCharacterEnabled = false;
                //Assert an END during the transfer of the last byte of data
                mbSession.SendEndEnabled = true;

                //Send a *CLS
                testClear(mbSession);
                //Now try a few commands - first *IDN?
                //Note that the \n is not included in these strings
                mbSession.Write("LANG NATIVE");
                mbSession.Write("*IDN?");
                Console.WriteLine("Response to *IDN?:");
                testStatus_Read(mbSession);
                //OID returns an instrument ID string
                mbSession.Write("OID");
                Console.WriteLine("Response to OID:");
                testStatus_Read(mbSession);
                //*OPT? returns the installed instrument options
                mbSession.Write("*OPT?");
                Console.WriteLine("Response to *OPT?:");
                testStatus_Read(mbSession);
                //This command doesn't exist,check that we catch it
                mbSession.Write("ABC");
                Console.WriteLine("Test ABC:");
                testStatus(mbSession);
                //this command is OK, no response is expected
                mbSession.Write("CH3");
                Console.WriteLine("Test CH3:");
                testStatus(mbSession);
```

```
            //Return to Local Control
            mbSession.Write("RTL\n");
            //Close the Session
            mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
            }
            keepConsoleUp();
        }
        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
        }
        //This function tests the Service Request Status Register
        //and does a read if MAV is set.
        //It prints an error message if any error is reported.
        private static void testStatus_Read(MessageBasedSession mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();
            string responseString = null;
            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                responseString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + responseString);
            }
            else if (((int)sb & b4) != 0)
            {
                responseString = mbSession.ReadString();
                Console.WriteLine(responseString);
            }
            mbSession.Write("*CLS\n");
        }
        //This function tests the Service Request Status Register.
        //It prints an error message if any error is reported.
        private static void testStatus(MessageBasedSession mbSession)
        {
            //These are the bits to check
            int b2 = 4;     //Error Queue is not empty
            //Read the Status Byte of Service Request Status Register
```

```
            //wait 50 ms
            Thread.Sleep(50);
            //then check status
            StatusByteFlags sb = mbSession.ReadStatusByte();
            string responseString = null;
            if (((int)sb & b2) != 0)
            {
                responseString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + responseString);
            }
            else
            {
                Console.WriteLine("OK");
            }
            mbSession.Write("*CLS\n");
        }
        private static void keepConsoleUp()
        {
            Console.WriteLine("");
            Console.WriteLine("Enter to Continue");
            Console.ReadLine();
        }
    }
}
```

Example 2 Discussion

Status Byte Checks

1. This example checks the Status Byte prior to reading the response. The Status Byte is the Service Request Status Register and is discussed in "Status Group Reporting" on page 2-29. LANG is set to NATIVE to use the MS464xB Series VNA Status Register Configuration. In general, a Message Available (MAV) response is expected when data is available to read. Otherwise, something else is expected to be in the Error Queue (ERRQ). Notice when the code sends an unknown command "ABC", the `testStatus()` function catches it and outputs the message in the Error Queue.
2. The response is slightly changed from Native to Lightning. If LANG LIGHT is set, then the Lightning configuration of the Status Register is used.

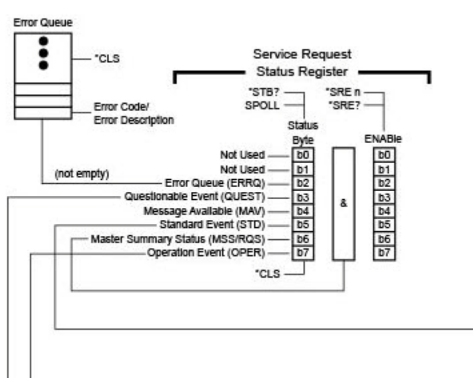


Figure D-25. The Service Request Status Register

Expected Output

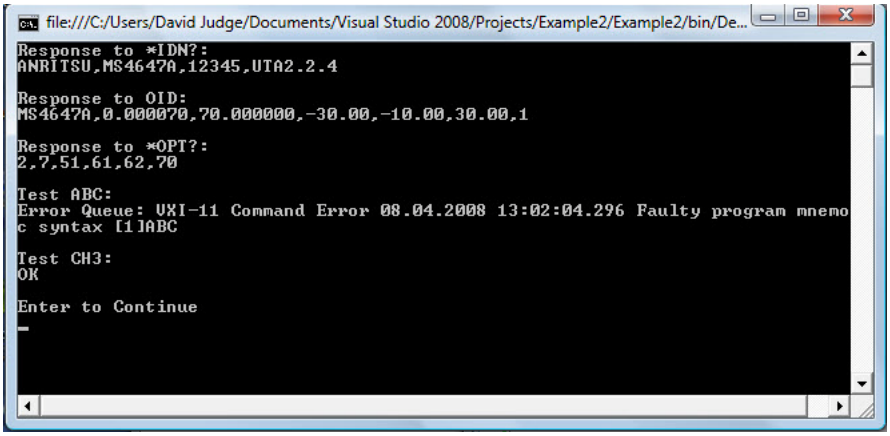


Figure D-26. Error message after sending the “ABC” command

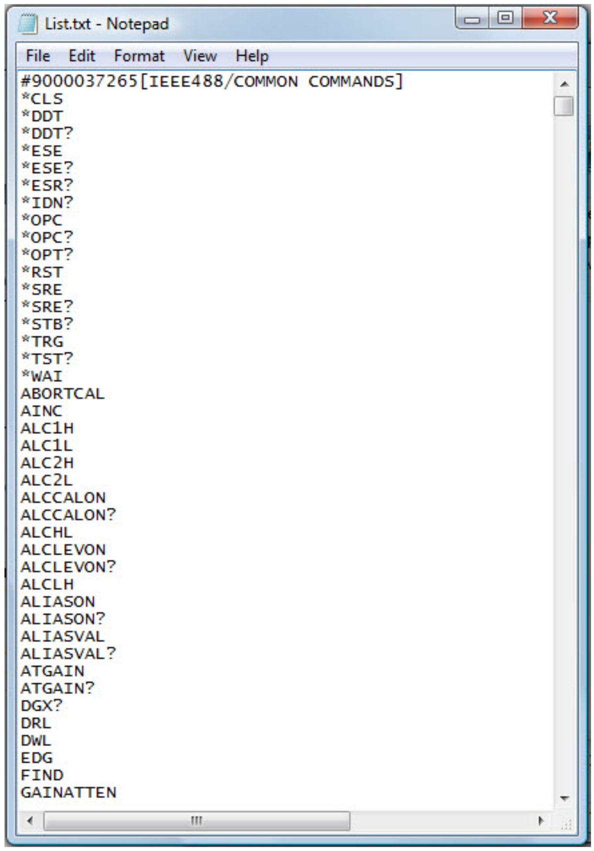
Example 3 – Sending Data to a File with the LIST Command

```
using System;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;
namespace Example3
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            string responseString = null;
            string[] split = null;
            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Timeout to 1 second (1000 ms)
                mbSession.Timeout = 1000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;
                //We're expected a lot of response data to the LIST command
                mbSession.DefaultBufferSize = 50000;
                testClear(mbSession);
                //Set the Language to NATIVE and send the LIST command
                mbSession.Write("LANG NATIVE\n");
                mbSession.Write("LIST\n");
                responseString = testStatus_Read(mbSession);
                //the results are delimited by a newline (\n)
                split = responseString.Split('\n');
                //Send results to a file
                StreamWriter output = new StreamWriter("List.txt");
                foreach (string s in split)
                    output.WriteLine(s);
                output.Close();
                mbSession.Write("RTL\n");
                mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
```

```
            }
            keepConsoleUp();
        }
        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
        }
       private static void keepConsoleUp()
        {
            Console.WriteLine("");
            Console.WriteLine("Enter to Continue");
            Console.ReadLine();
        }
        //This function tests the Service Request Status Register
        //and does a read if MAV is set.
        //It prints an error message if any error is reported.
        private static string testStatus_Read(MessageBasedSession mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();
            string responseString = null;
            string errorString = null;
            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                errorString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + errorString);
            }
            else if (((int)sb & b4) != 0)
            {
                responseString = mbSession.ReadString();
            }
            mbSession.Write("*CLS\n");
            return responseString;
        }
    }
}
```

Example 3 Discussion

1. This example and the next few examples do not write anything to the Console unless there is an error. Output is sent to a file in the same directory of the running program (\Example3\Bin\Debug\List.txt):



List.txt - Notepad

File Edit Format View Help

```
#9000037265[IEEE488/COMMON COMMANDS]
*CLS
*DDT
*DDT?
*ESE
*ESE?
*ESR?
*IDN?
*OPC
*OPC?
*OPT?
*RST
*SRE
*SRE?
*STB?
*TRG
*TST?
*WAI
ABORTCAL
AINC
ALC1H
ALC1L
ALC2H
ALC2L
ALCCALON
ALCCALON?
ALCHL
ALCLEVON
ALCLEVON?
ALCLH
ALIASON
ALIASON?
ALIASVAL
ALIASVAL?
ATGAIN
ATGAIN?
DGX?
DRL
DWL
EDG
FIND
GAINATTEN
```

Figure D-27. List of all commands supported by the MS464xB Series VNA

Use of LANG LIGHT

2. Use LANG LIGHT – Lightning has the same command, but is comma delimited:

```
//Set the Language to LIGHTNING and send the LIST command
mbSession.Write("LANG LIGHT\n");
mbSession.Write("LIST\n");
responseString = testStatus_Read(mbSession);
//the results are delimited by a comma
split = responseString.Split(',');
```

Anritsu GPIB, USB, VXI-11 Exerciser

3. Use the Anritsu GPIB, USB, VXI-11 Exerciser to get more help on any command. Help will tell you what type of command (Native, Lightning, HP8510) and provides syntax.

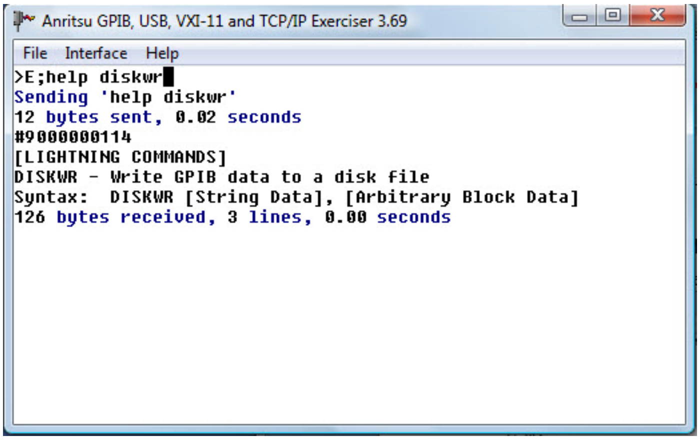


Figure D-28. Anritsu GPIB, USB, VXI-11 Exerciser

Example 4 – Acquiring ASCII Data, Arbitrary Block

```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;
namespace Example4
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            string responseStringN = null;
            string responseStringL = null;
            string[] splitN = null;
            string[] splitL = null;
            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Timeout to 1 second (1000 ms)
                mbSession.Timeout = 1000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;
                //We'll limit the number of data points to keep buffer small
                mbSession.DefaultBufferSize = 5000;
                testClear(mbSession);
                //Set the Language to NATIVE
                mbSession.Write("LANG NATIVE\n");
                mbSession.Write(":SENSE:SWEEP:POINTS 5\n");
                mbSession.Write("CH1;SMI;\n");
                mbSession.Write("TRS;WFS;HLD\n");
                mbSession.Write("FMA;OFD\n");
                responseStringN = testStatus_ReadArb(mbSession);
                //the results are delimited by a newline (\n)
                splitN = responseStringN.Split('\n');
                //Set the Language to LIGHTNING
                mbSession.Write("LANG LIGHT\n");
                mbSession.Write("FMA;OFD\n");
                responseStringL = testStatus_ReadArb(mbSession);
                //the results are delimited by a comma
                splitL = responseStringL.Split(',');
                //Send results to a file
                StreamWriter output = new StreamWriter("OFD.txt");
                output.WriteLine("Native Results");
                foreach (string s in splitN)
                    output.WriteLine(s);
```

```
            output.WriteLine("Lightning Results");
            foreach (string s in splitL)
                output.WriteLine(s);
            output.Close();
            mbSession.Write("RTL\n");
            mbSession.Dispose();
        }
        catch (VisaException v_exp)
        {
            Console.WriteLine("Visa caught an error!!");
            Console.WriteLine(v_exp.Message);
        }
        catch (Exception exp)
        {
            Console.WriteLine("Something didn't work!!");
            Console.WriteLine(exp.Message);
        }
    }
    private static void testClear(MessageBasedSession mbSession)
    {
        mbSession.Write("*CLS\n");
    }
    private static string testStatus_ReadArb(MessageBasedSession mbSession)
    {
        //These are the bits to check
        int b2 = 4,     //Error Queue is not empty
            b4 = 16;    //MAV = Message Available
        string responseString = null;
        string replyString = null;
        string errorString = null;
        //Read the Status Byte of Service Request Status Register
        StatusByteFlags sb = mbSession.ReadStatusByte();

        while (((int)sb & (b2 + b4)) == 0)
        {
            Thread.Sleep(10);
            sb = mbSession.ReadStatusByte();
        }
        if (((int)sb & b2) != 0)
        {
            errorString = mbSession.Query("OGE\n");
            Console.WriteLine("Error Queue: " + errorString);
        }
        else if (((int)sb & b4) != 0)
        {
            responseString = mbSession.ReadString();
            replyString = stripHeader(responseString);

        }
        mbSession.Write("*CLS\n");
        return replyString;
    }
    private static string stripHeader(string responseString)
    {
        int i = 0;
```

```
                string strReturn = null;
                if (responseString[i++] == '#')
                {
                    //Header is ASCII, get 2nd byte and convert to int
                   StringBuilder sCount = new StringBuilder(responseString, i++, 1, 1);
                    int count1 = int.Parse(sCount.ToString());
                    //now read the bytecount string and convert to int
                    StringBuilder sBytes = new
StringBuilder(responseString,i,count1,count1);
                    int count2 = int.Parse(sBytes.ToString());
                    //set the index of the start of the data
                    i += count1;
                    //return the string with the header stripped off
                    strReturn = responseString.Substring(i);
                }
                return strReturn;
            }
        }
    }
```

Example 4 Discussion

Output Results

1. Output result values are the same for Native mode and Lightning mode (since the unit is in HOLD), but the delimiters are different. Native mode uses a comma delimiters to separate the two values at each frequency and a newline (\n) to separate pairs of values. Lightning mode separates each value with a comma. The display was set for Smith Chart, impedance.

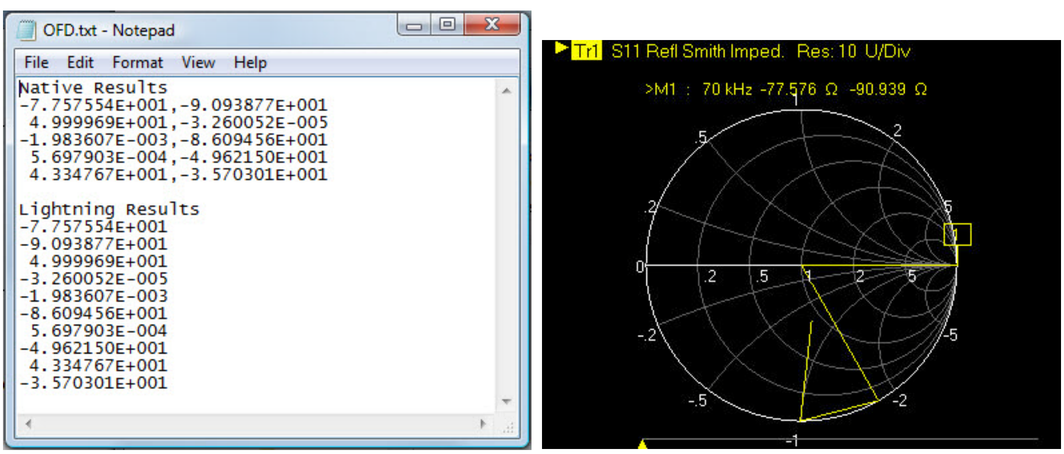


Figure D-29. OFD Output File and Trace Display

OFD Command

2. The OFD command returns an arbitrary block containing either ASCII or Binary data depending on the currently selected format. This example selects ASCII using the FMA command. The Arbitrary block header is stripped off with the `stripHeader(string responseString)` function. The next example shows how to get binary data.

Example 5 – Acquiring Binary Data, Arbitrary Block

```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;
namespace Example5
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            double[] responseArray = null;
            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Timeout to 1 second (1000 ms)
                mbSession.Timeout = 1000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;
                //We'll limit the number of data points to keep buffer small
                mbSession.DefaultBufferSize = 5000;
                testClear(mbSession);
                //Set the Language to NATIVE
                mbSession.Write("LANG NATIVE\n");
                mbSession.Write("CH2;MPH;\n");
                mbSession.Write(":SENSE:SWEEP:POINTS 8\n");
                mbSession.Write("TRS;WFS;HLD\n");
                mbSession.Write("LSB;FMB;OFD\n");
                responseArray = testStatus_ReadArbBinaryDouble(mbSession);

                //Send results to a file
                StreamWriter output = new StreamWriter("OFD.txt");
                output.WriteLine("Native Results (Mag-Phase)");
                foreach (double d in responseArray)
                    output.WriteLine(d);
                output.Close();
                mbSession.Write("RTL\n");
                mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
```

```
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
            }
        }
        private static double[] testStatus_ReadArbBinaryDouble(MessageBasedSession 
  mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            byte[] responsebytes = null;
            double[] replybytes = null;
            string errorString = null;
            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();
            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                errorString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + errorString);
            }
            else if (((int)sb & b4) != 0)
            {
                responsebytes = mbSession.ReadByteArray();
                replybytes = arbToDouble(responsebytes);
            }
            mbSession.Write("*CLS\n");
            return replybytes;
        }
        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
        }
        //Here we convert Arb Block Binary Data to a double array
        private static double[] arbToDouble(byte[] responseBytes)
        {
            int i = 0;
            double[] dReturn = null;
            //Arbitrary Block should start with a #
            if (responseBytes[i++] == '#')
            {
                //Header is ASCII, get 2nd byte and convert to int
                string sCount = ASCIIEncoding.ASCII.GetString(responseBytes, i++, 
  1);
                int count1 = int.Parse(sCount);
                //now read the bytecount string and convert to int
                string sBytes = ASCIIEncoding.ASCII.GetString(responseBytes, i, 
  count1);
                int count2 = int.Parse(sBytes);
                //the number of doubles is the #bytes/sizeof(double)
                int dataCount = count2 / sizeof(double);
```

```
                    //resize the response array
                    dReturn = new double[dataCount];
                    //set the index of the start of the data
                    i += count1;
                    //There are many ways to convert a byte array to a double array
                    MemoryStream stream = new MemoryStream(responseBytes, i, count2);
                    //BinaryReader reads this data type in little-endian format
                    //So we must use the LSB mnemonic when acquiring the data
                    BinaryReader reader = new BinaryReader(stream);
                    for (int ii = 0; ii < dataCount; ii++)
                    {
                        dReturn[ii] = reader.ReadDouble();
                    }
                 }
                return dReturn;
            }
        }
    }
```

Example 5 Discussion

Output File

1. Output file from this program should be in \Example5\bin\Debug\OFD.txt:

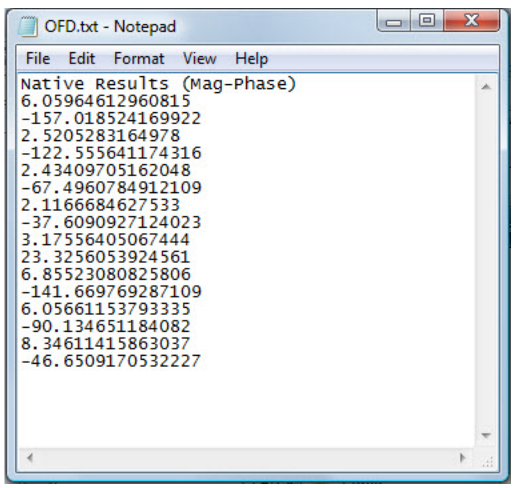


Figure D-30. Binary Output, Arbitrary Block

Example 6 – Output S2P File

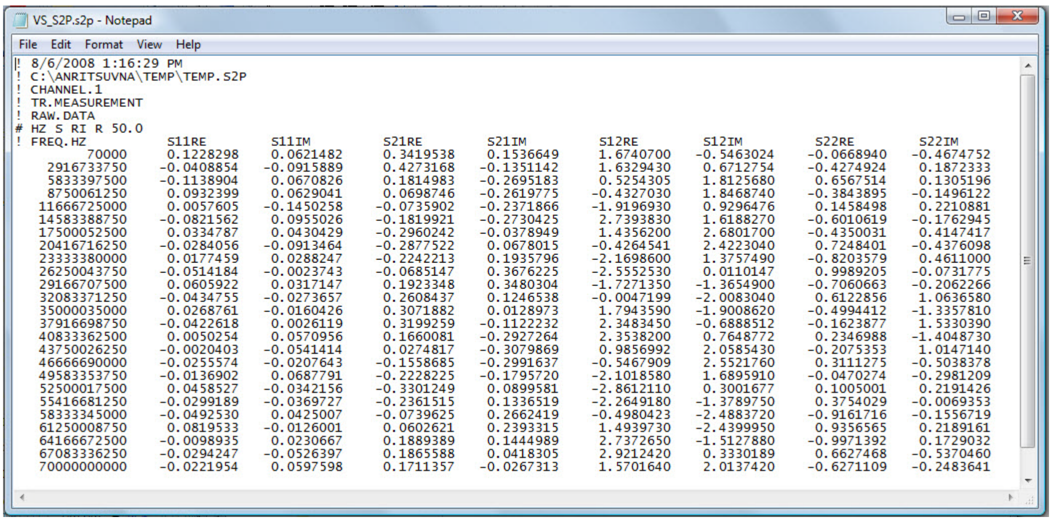
```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;
namespace Example7
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            string responseStringN = null;
            string[] splitN = null;
            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Timeout to 3 second (1000 ms)
                mbSession.Timeout = 3000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;
                //We'll limit the number of data points to keep buffer small
                mbSession.DefaultBufferSize = 5000;
                testClear(mbSession);
                //Set the Language to NATIVE
                mbSession.Write("LANG NATIVE\n");
                mbSession.Write(":SENSE:SWEEP:POINTS 25\n");
                //set format to Hz and Re/Im
                mbSession.Write(":FORM:SNP:FREQ HZ\n");
                mbSession.Write(":FORM:SNP:PAR REIM\n");
                //Output S2P file over GPIB
                mbSession.Write("TRS;WFS;OS2P\n");
                responseStringN = testStatus_ReadArb(mbSession);
                //the results are delimited by a newline (\n)
                splitN = responseStringN.Split('\n');
                //Send results to a file
                StreamWriter output = new StreamWriter("VS_S2P.s2p");
                foreach (string s in splitN)
                    output.WriteLine(s);
                output.Close();
                mbSession.Write("RTL\n");
                mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
```

```
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
            }
        }
        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
        }
        private static string testStatus_ReadArb(MessageBasedSession mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            string responseString = null;
            string replyString = null;
            string errorString = null;
            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();
            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                errorString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + errorString);
            }
            else if (((int)sb & b4) != 0)
            {
                responseString = mbSession.ReadString();
                replyString = stripHeader(responseString);
            }
            mbSession.Write("*CLS\n");
            return replyString;
        }
        private static string stripHeader(string responseString)
        {
            int i = 0;
            string strReturn = null;
            if (responseString[i++] == '#')
            {
                //Header is ASCII, get 2nd byte and convert to int
               StringBuilder sCount = new StringBuilder(responseString, i++, 1, 1);
                int count1 = int.Parse(sCount.ToString());
                //now read the bytecount string and convert to int
               StringBuilder sBytes = new StringBuilder(responseString, i, count1,
count1);
                int count2 = int.Parse(sBytes.ToString());
```

```
                    //set the index of the start of the data
                    i += count1;
                    //return the string with the header stripped off
                    strReturn = responseString.Substring(i);
                }
                return strReturn;
            }
        }
    }
```

Example 6 Discussion

Expected output is in \Example7\Bin\Debug\VS_S2P.s2p



```
VS_S2P.s2p - Notepad
File Edit Format View Help
! 8/6/2008 1:16:29 PM
! C:\ANRITSUVNA\TEMP\TEMP.S2P
! CHANNEL.1
! TR.MEASUREMENT
! RAW.DATA
# HZ S RI R 50.0
! FREQ.HZ        S11RE      S11IM      S21RE      S21IM      S12RE      S12IM      S22RE      S22IM
      70000   0.1228298  0.0621482  0.3419538  0.1536649  1.6740700 -0.5463024 -0.0668940 -0.4674752
 2916733750  -0.0408854 -0.0915889  0.4273168 -0.1351142  1.6329430  0.6712754 -0.4274924  0.1872333
 5833397500  -0.1138904  0.0670826  0.1814983 -0.2695183  0.5254305  1.8125680  0.6567514  0.1305196
 8750061250   0.0932399  0.0629041  0.0698746 -0.2619775 -0.4327030  1.8468740 -0.3843895 -0.1496122
11666725000   0.0057605 -0.1450258 -0.0735902 -0.2371866 -1.9196930  0.9296476  0.1458498  0.2210881
14583388750  -0.0821562  0.0955026 -0.1819921 -0.2730425  2.7393830  1.6188270 -0.6010619 -0.1762945
17500052500   0.0334787  0.0430429 -0.2960242 -0.0378949  1.4356200  2.6801700 -0.4350031  0.4147417
20416716250  -0.0284056 -0.0913464 -0.2877522  0.0678015 -0.4264541  2.4223040  0.7248401 -0.4376098
23333380000   0.0177459  0.0288247 -0.2242213  0.1935796 -2.1698600  1.3757490 -0.8203579  0.4611000
26250043750  -0.0514184 -0.0023743 -0.0685147  0.3676225 -2.5552530  0.0110147  0.9989205 -0.0731775
29166707500   0.0605922  0.0317147  0.1923348  0.3480304 -1.7271350 -1.3654900 -0.7060663 -0.2062266
32083371250  -0.0434755 -0.0273657  0.2608437  0.1246538 -0.0047199 -2.0083040  0.6122856  1.0636580
35000035000   0.0268761 -0.0160426  0.3071882  0.0128973  1.7943590 -1.9008620 -0.4994412 -1.3357810
37916698750  -0.0422618  0.0026119  0.3199259 -0.1122232  2.3483450 -0.6888512 -0.1623877  1.5330390
40833362500   0.0050254  0.0570956  0.1660081 -0.2927264  2.3538200  0.7648772  0.2346988 -1.4048730
43750026250  -0.0020403 -0.0541414  0.0274817 -0.3079869  0.9856992  2.0585430 -0.2075353  1.0147140
46666690000  -0.0255574 -0.0207643 -0.1558685 -0.2991637 -0.5467909  2.5521760  0.3111275 -0.5038378
49583353750  -0.0136902  0.0687791 -0.2228225 -0.1795720 -2.1018580  1.6895910 -0.0470274 -0.2981209
52500017500   0.0458527 -0.0342156 -0.3301249  0.0899581 -2.8612110  0.3001677  0.1005001  0.2191426
55416681250  -0.0299189 -0.0369727 -0.2361515  0.1336519 -2.2649180 -1.3789750  0.3754029 -0.0069353
58333345000  -0.0492530  0.0425007 -0.0739625  0.2662419 -0.4980423 -2.4883720 -0.9161716 -0.1556719
61250008750   0.0819533 -0.0126001  0.0602621  0.2393315  1.4939730 -2.4399950  0.9356565  0.2189161
64166672500  -0.0098935  0.0230667  0.1889389  0.1444989  2.7372650 -1.5127880 -0.9971392  0.1729032
67083336250  -0.0294247 -0.0526397  0.1865588  0.0418305  2.9212420  0.3330189  0.6627468 -0.5370460
70000000000  -0.0221954  0.0597598  0.1711357 -0.0267313  1.5701640  2.0137420 -0.6271109 -0.2483641
```

Figure D-31. Transfer of an S2P file to the PC

Example 7 – Output a Bitmap

```
using System;
using System.Text;
using System.IO;
using NationalInstruments.VisaNS;
using System.Threading;
namespace Example6
{
    class Program
    {
        static void Main(string[] args)
        {
            //VXI-11 Connection string
            string sAddress = "TCPIP0::10.0.1.196::INSTR";
            //The VNA uses a message based session
            MessageBasedSession mbSession = null;
            //But we'll just open a generic Session first
            Session mySession = null;
            byte[] responseArray = null;
            try
            {
                //open a Session to the VNA
                mySession = ResourceManager.GetLocalManager().Open(sAddress);
                //cast this to a message based session
                mbSession = (MessageBasedSession)mySession;
                //Timeout to 3 seconds (3000 ms)
                mbSession.Timeout = 3000;
                mbSession.TerminationCharacter = 0x0a;
                mbSession.TerminationCharacterEnabled = false;
                mbSession.SendEndEnabled = true;
                //We expect about 534k bytes
                mbSession.DefaultBufferSize = 600000;
                testClear(mbSession);
                //Set the Language to LIGHT,
                mbSession.Write("LANG LIGHT\n");
                //set color on white and output bitmap over GPIB
                mbSession.Write("BMPC;OBMP\n");
                responseArray = testStatus_ReadArbBinaryByte(mbSession);
                //Send results to a new Binary file
                BinaryWriter output = new BinaryWriter(File.Open("VS.bmp",
FileMode.Create));
                output.Write(responseArray);
                output.Close();
                mbSession.Write("RTL\n");
                mbSession.Dispose();
            }
            catch (VisaException v_exp)
            {
                Console.WriteLine("Visa caught an error!!");
                Console.WriteLine(v_exp.Message);
            }
            catch (Exception exp)
            {
                Console.WriteLine("Something didn't work!!");
                Console.WriteLine(exp.Message);
            }
```

```
        }
        private static byte[] testStatus_ReadArbBinaryByte(MessageBasedSession
mbSession)
        {
            //These are the bits to check
            int b2 = 4,     //Error Queue is not empty
                b4 = 16;    //MAV = Message Available
            byte[] responsebytes = null;
            byte[] replybytes = null;
            string errorString = null;
            //Read the Status Byte of Service Request Status Register
            StatusByteFlags sb = mbSession.ReadStatusByte();
            while (((int)sb & (b2 + b4)) == 0)
            {
                Thread.Sleep(10);
                sb = mbSession.ReadStatusByte();
            }
            if (((int)sb & b2) != 0)
            {
                errorString = mbSession.Query("OGE\n");
                Console.WriteLine("Error Queue: " + errorString);
            }
            else if (((int)sb & b4) != 0)
            {
                responsebytes = mbSession.ReadByteArray();
                replybytes = arbToByte(responsebytes);
            }
            mbSession.Write("*CLS\n");
            return replybytes;
        }
        private static void testClear(MessageBasedSession mbSession)
        {
            mbSession.Write("*CLS\n");
        }
        //Convert Arb Block Binary Data to Byte array
        private static byte[] arbToByte(byte[] responseBytes)
        {
            int i = 0;
            byte[] dReturn = null;
            //Arbitrary Block should start with a #
            if (responseBytes[i++] == '#')
            {
                //Header is ASCII, get 2nd byte and convert to int
                string sCount = ASCIIEncoding.ASCII.GetString(responseBytes, i++,
1);
                int count1 = int.Parse(sCount);
                //now read the bytecount string and convert to int
                string sBytes = ASCIIEncoding.ASCII.GetString(responseBytes, i,
count1);
                int count2 = int.Parse(sBytes);
                //the number of doubles is the #bytes/sizeof(double)
                int dataCount = count2 / sizeof(byte);
                //resize the response array
                dReturn = new byte[dataCount];
                //set the index of the start of the data
                i += count1;
```

```
                MemoryStream stream = new MemoryStream(responseBytes, i, count2);
                //BinaryReader reads this data type in little-endian format
                //So we must use the LSB mnemonic when acquiring the data
                BinaryReader reader = new BinaryReader(stream);
                for (int ii = 0; ii < dataCount; ii++)
                {
                    dReturn[ii] = reader.ReadByte();
                }
            }
            return dReturn;
        }
    }
}
```

Example 7 Discussion

The output should be in \Example6\Bin\Debug\VS.bmp

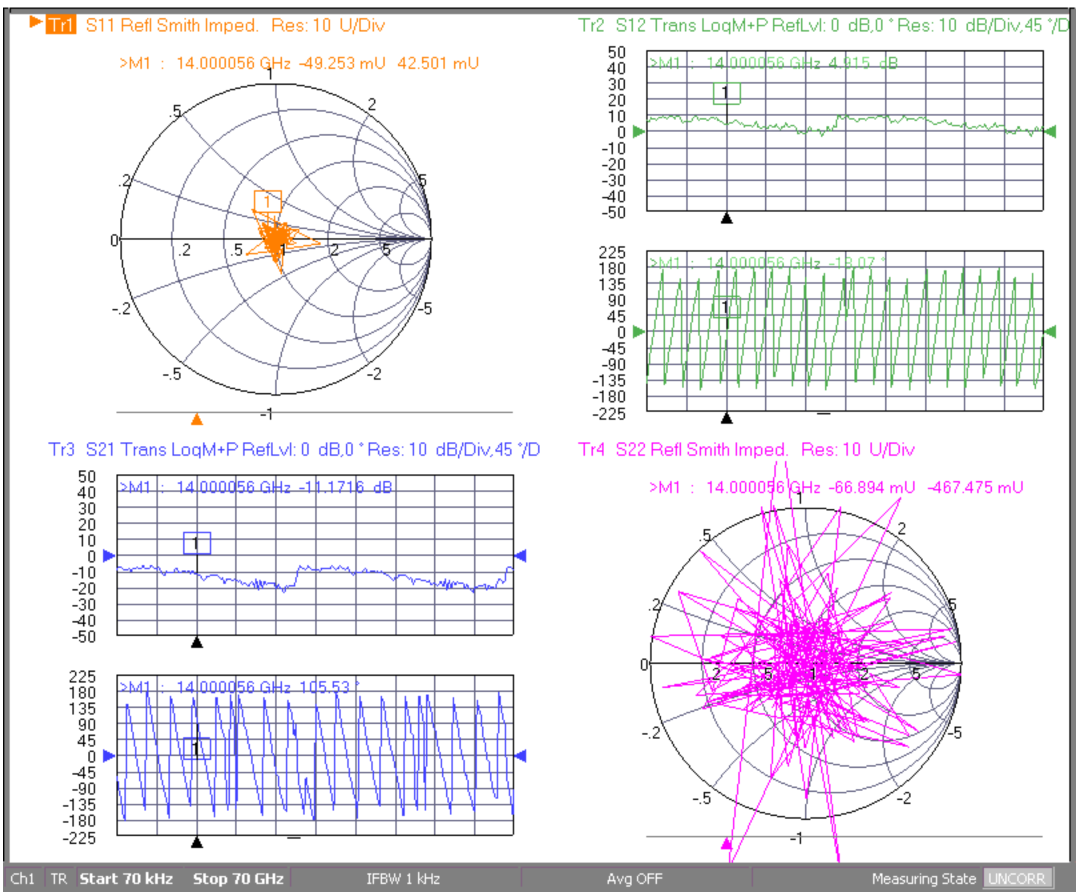


Figure D-32. Results from Example 6

Property	Value
Image	
Dimensions	804 x 663
Width	804 pixels
Height	663 pixels
Bit depth	8

Figure D-33. Bitmap Properties

D-7 LabView Programming Examples

This section describes eight programs that demonstrate the use of the 37XXX LabVIEW driver controlling the MS464xB Series VNA. The programs provide functions to:

- Open a session
- Send the *IDN? command
- Check for errors
- Send data to a file with the LIST command
- Acquire trace data
- Acquire Smith Chart data
- Output an S2P File
- Output a BMP file

Setup Communications

1. Set up VXI-11 (TCP/IP) communications by noting the IP Address of the VNA and set up a resource (a connection string).

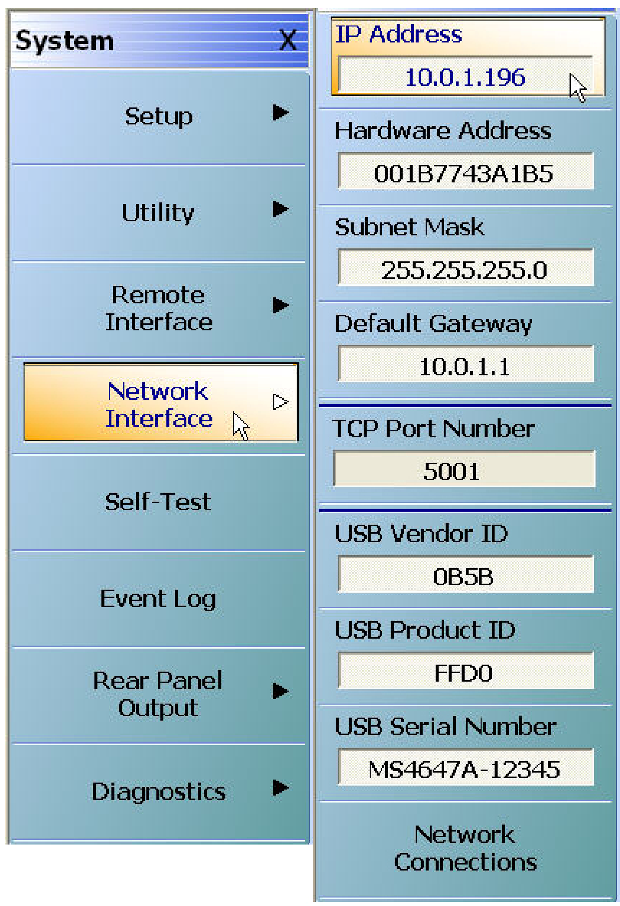


Figure D-34. MS464xB Series VNA IP Address

2. Set up the VISA resource using the Measurement and Automation Explorer (MAX).

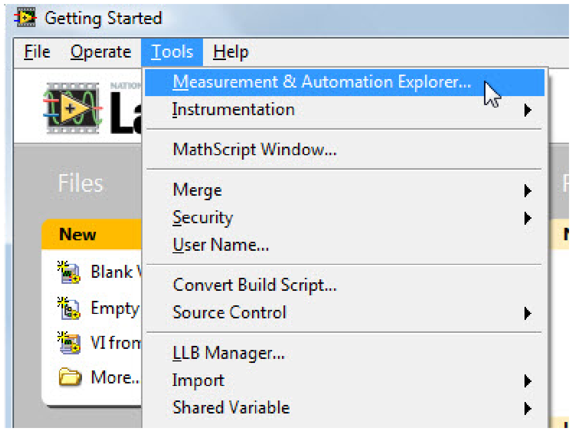


Figure D-35. LabView Tools

3. Create a new resource in MAX by right clicking Devices and Interfaces and select Create New.

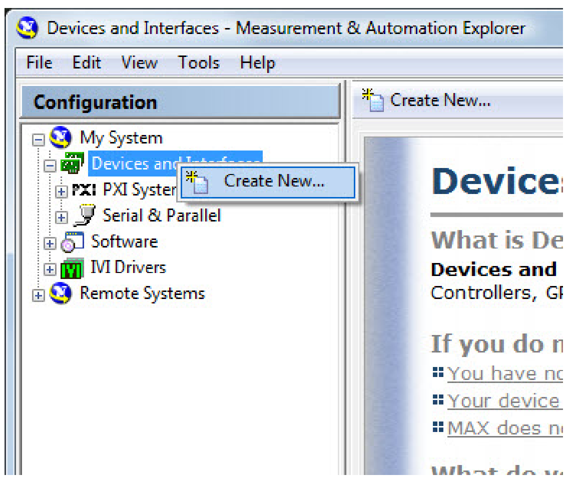


Figure D-36. Measurement and Automation Explorer (MAX)

4. Select the VISA TCP/IP Resource.

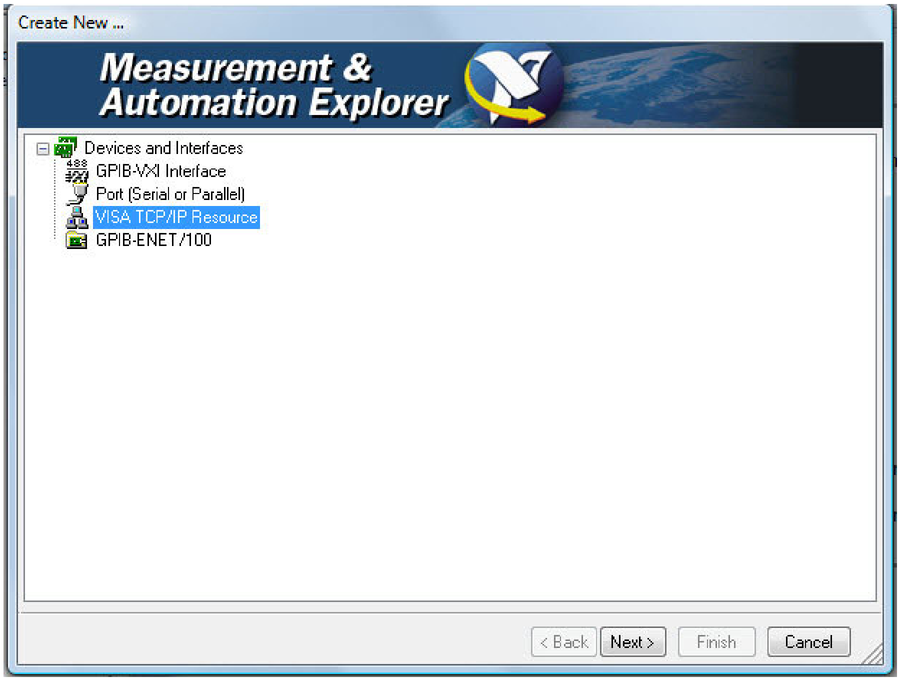


Figure D-37. Create New Resource Dialog

5. If the controlling PC and the VNA are on the same local sub-network (this is usually true if the first 3 numbers in the IP address are the same – for example, 10.0.1.x in this case), then you can Auto-detect the VNA. Otherwise, you need to manually enter the IP address.

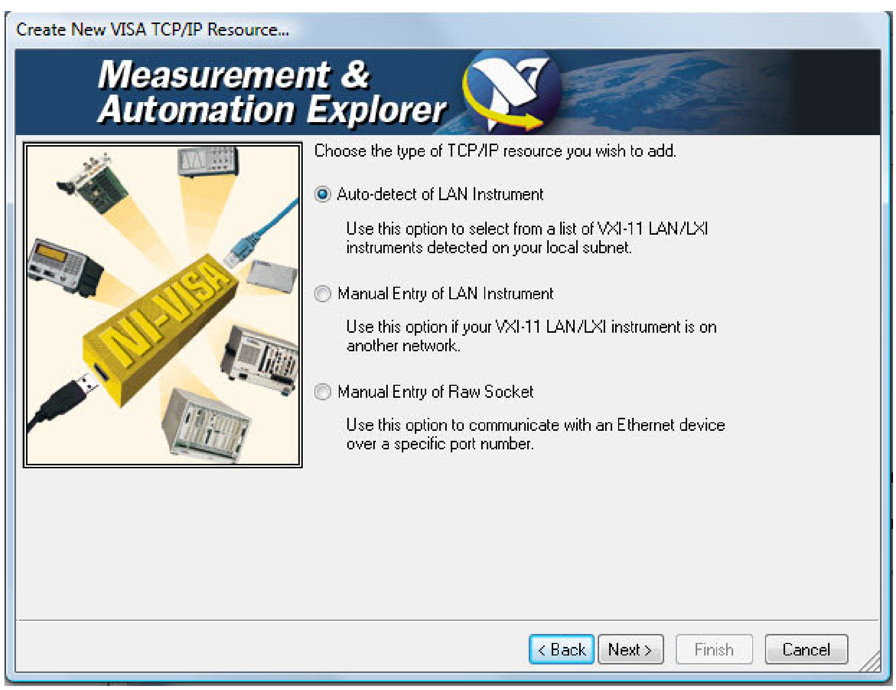


Figure D-38. Create New VISA TCP/IP Resource Dialog

6. Select the detected instrument.

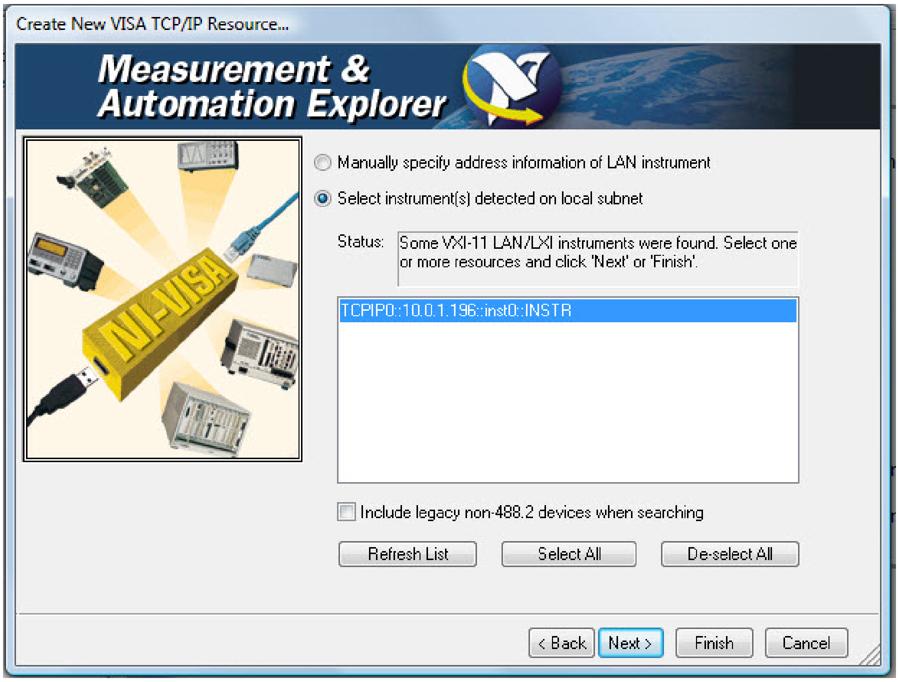


Figure D-39. Create New VISA TCP/IP Resource Dialog

7. Give the instrument an alias (the VectorStar_Test alias is used in LabVIEW).

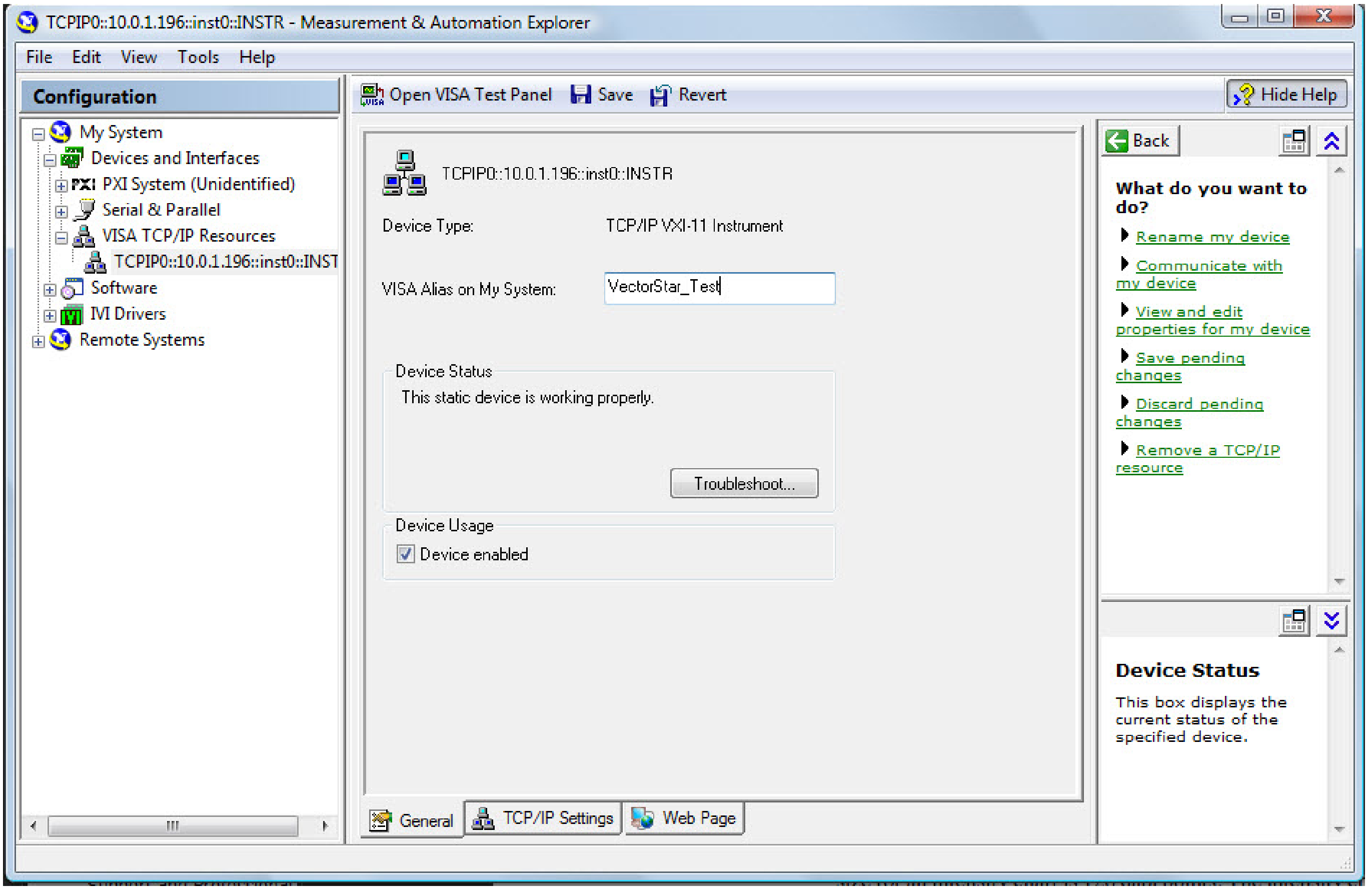


Figure D-40. Measurement and Automation Explorer (MAX)

8. Note that the VISA connection string has been replaced with the VISA alias.

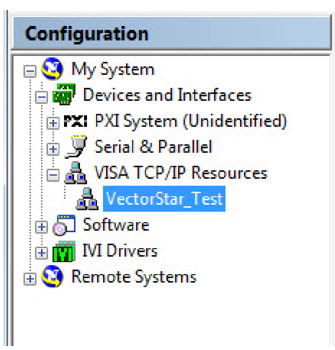


Figure D-41. Measurement and Automation Explorer Configuration

9. In LabVIEW, begin creating the first example, a New VI.

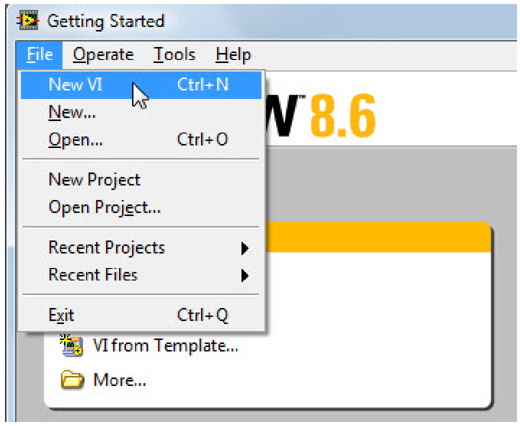


Figure D-42. LabVIEW New VI

Example 1 – Opening a Session

In this first example, a communication session is opened to the VNA and two of the driver VIs are used to get some information about the VNA. The block diagram below shows four VIs from the Anritsu 37XXX Series driver:

- Initialize.vi
- Revision Query.vi
- Instrument Options.vi
- Close.vi.

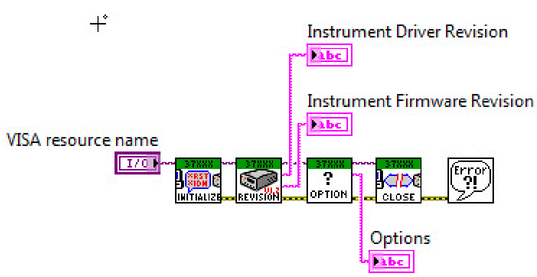


Figure D-43. VI Block Diagram

1. In the Front Panel dialog, select "VectorStar_Test", which was set up previously, as the VISA resource name.

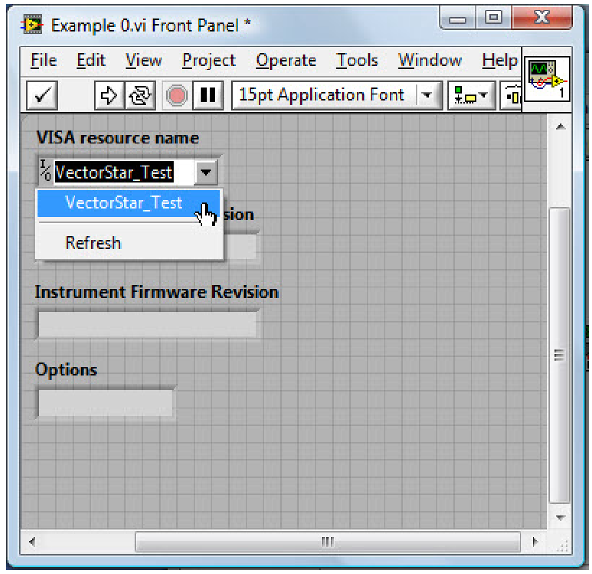


Figure D-44. Front Panel Dialog

2. After running this VI, the Instrument Driver Version, Instrument Firmware Version, and Options are filled in.

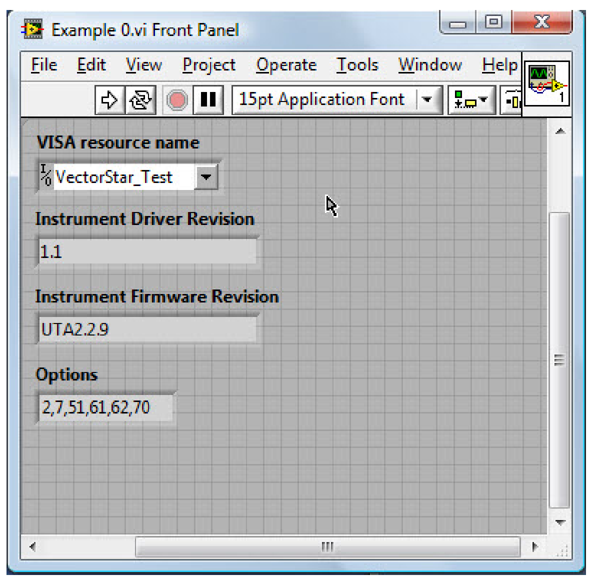


Figure D-45. Front Panel Dialog

Example 2 – Sending the *IDN? Command

The previous example used only driver VIs to get some information from the VNA. The GPIB command, "*IDN?" returns the Manufacturer, Model #, Serial Number and Firmware Version. In this example, the "*IDN?" command is used directly and the return string is parsed into the different parts of the response.

1. Directly send a GPIB string using the VISA Write function, then read the response from the VNA using the VISA Read function.

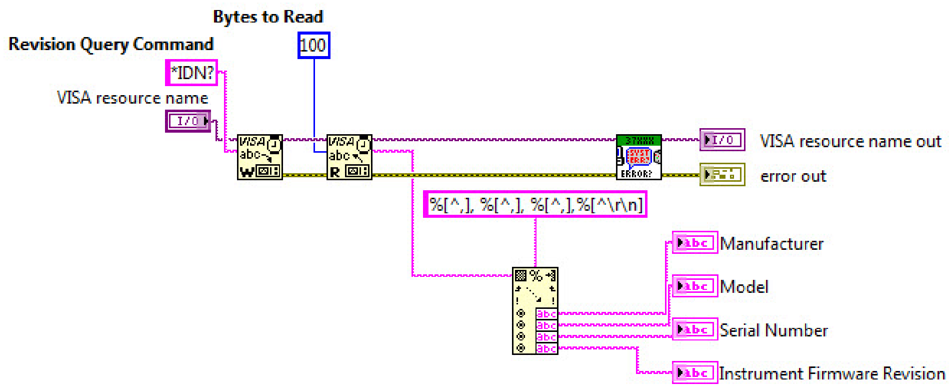


Figure D-46. VI Block Diagram

2. Use the Scan From String function to grab the different comma-separated values in the response string.

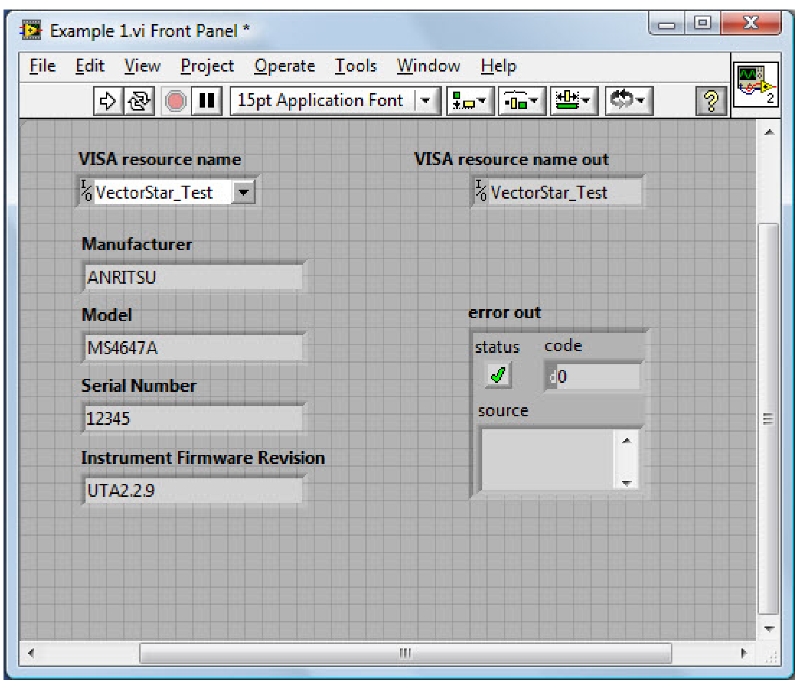


Figure D-47. Front Panel for Example 1

Example 2 – Error Checking

Most of the 37xxx driver VIs use the Error Query.vi. This example shows that if an invalid GPIB string is sent to the VNA, then the Error Query VI catches the error and displays the error message from the VNA.

1. Two valid strings, “*IDN?” and “OID,” are sent, but the third string is not a valid GPIB command and the instrument reports this to the Front Panel.
2. Mix VISA Writes and Reads with the 37XXX driver’s Error Query VI.

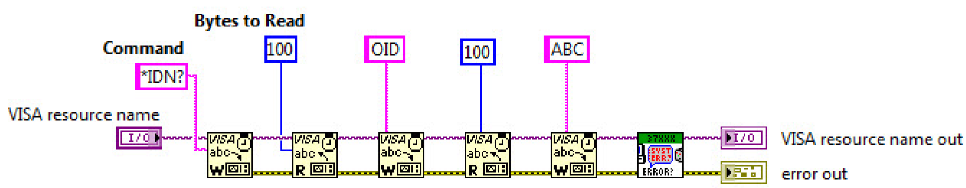


Figure D-48. VI Block Diagram

3. The “ABC” command sent is invalid and was captured by the Error Query VI.

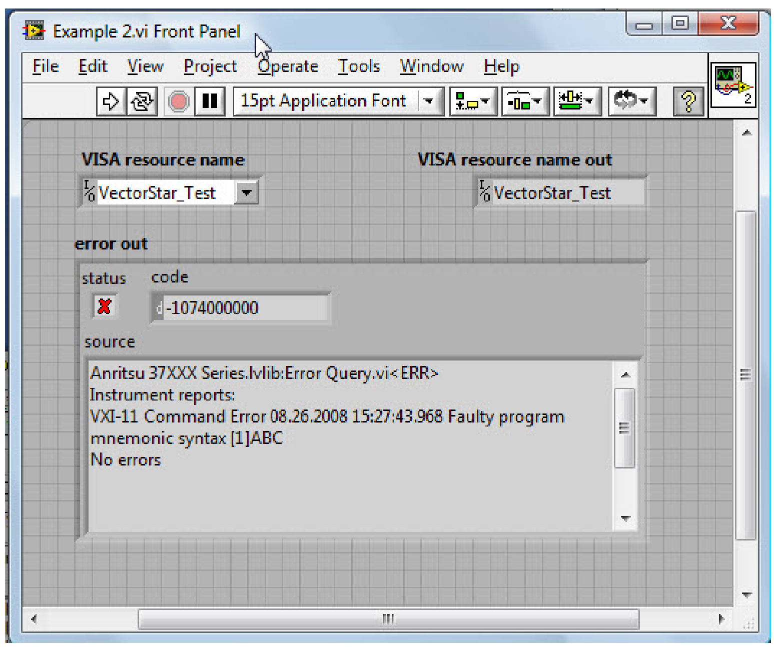


Figure D-49. Front Panel for Example 2

Example 3 – Sending Data to a File with the LIST Command

This example sends output to a file that the user may specify (a file dialog is opened). The VISA Write command is used to send the LIST command to the VNA. The "LANG NATIVE" command is used because this sends a newline after each command in the output. If "LANG LIGHT" is used, then the commands are separated by a comma and the file is more difficult to read.

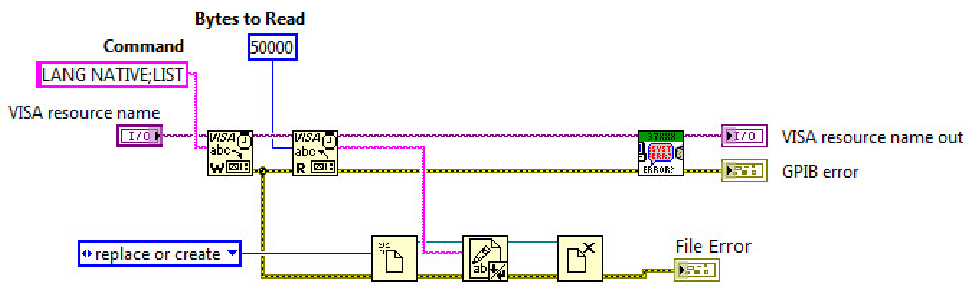


Figure D-50. VI Block Diagram

The Front Panel shows that no errors were produced with either the GPIB code or with the File writing code.

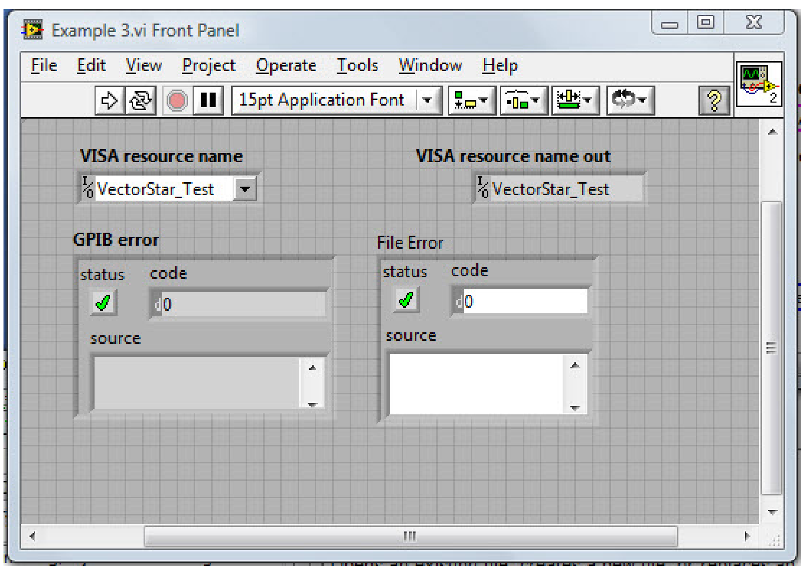


Figure D-51. Front Panel for Example 3

The List of all commands supported by the MS464xB Series VNA is output to a text file. Note that the first line has an arbitrary block header, which is covered in Example 6.

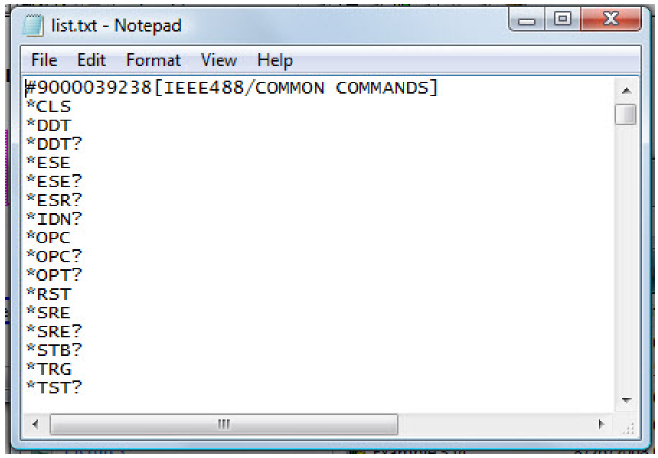


Figure D-52. Command List

Use the Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser to get more help on any command. Help displays the type of command (Lightning, Native, or HP8510) and provides syntax.

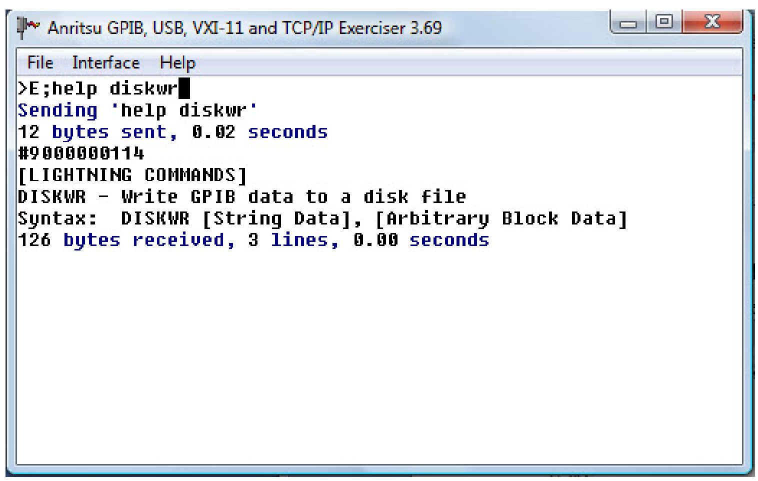


Figure D-53. Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser

Example 4 – Acquiring Trace Data

Trace 2 is set to Log magnitude and Phase data and the data comes out in a one dimensional interleaved array. The array must be parsed to get the data into two separate arrays. In this example, the VNA is initialized to get Log Magnitude and Phase Data (Final Data) from Trace 2. When two sets of data exist, the 1D array must be parsed into two arrays since data 0, 2, 4, etc., is Log Magnitude data and data 1, 3, 5, etc., is phase data.

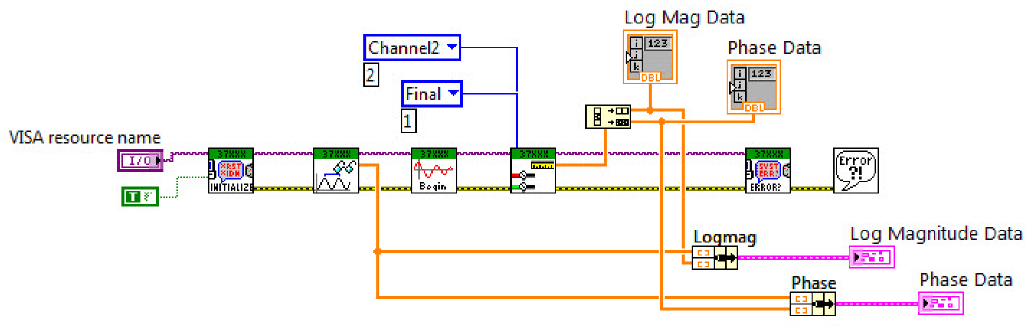


Figure D-54. VI Block Diagram

Send separate traces to each XY Graph. (Note that this data is from a simulated MS464xB and might not look authentic).

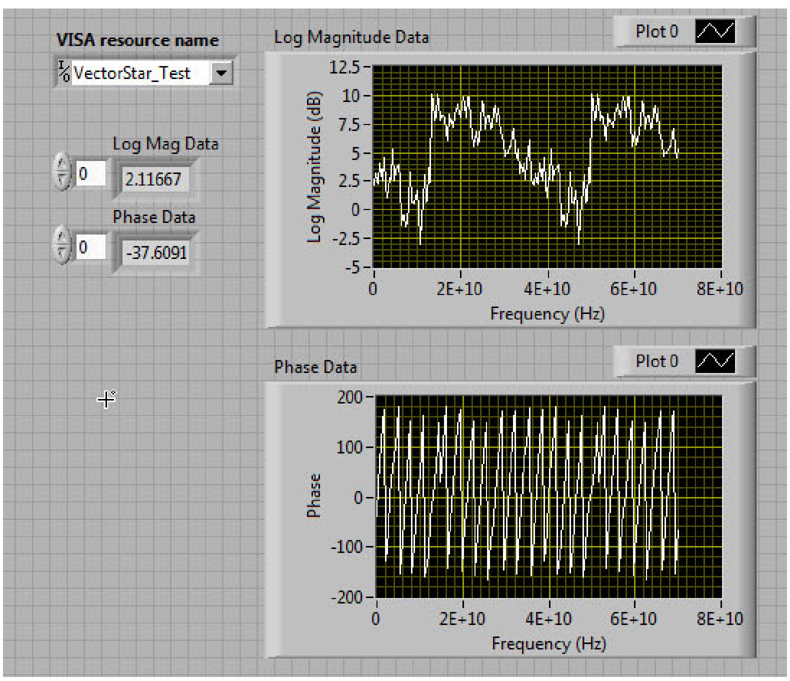


Figure D-55. Front Panel for Example 4

Example 5 – Smith Chart Data

This example is similar to the previous example except the instrument is **not** reset, which sets up smith charts on trace 1 and 4. Although the frequency data is not needed to display the smith chart, the frequency list is extracted from the Read Frequency Values VI and the data count is determined from the number of frequencies. This data count is passed to the For Loop control.

By default Trace1 is set to output impedance values. Smith chart takes a normalized impedance (normalized to 1), so we divide the impedance/reactance pairs by 50 (ohms) to get normalized smith chart data.

Use the Frequency Array to get the number of points to input into the For Loop control. The Channel Data from Channel 1 is Impedance/Reactance, so divide by Z0 (or 50 ohms, hardcoded here) to get the smith chart data.

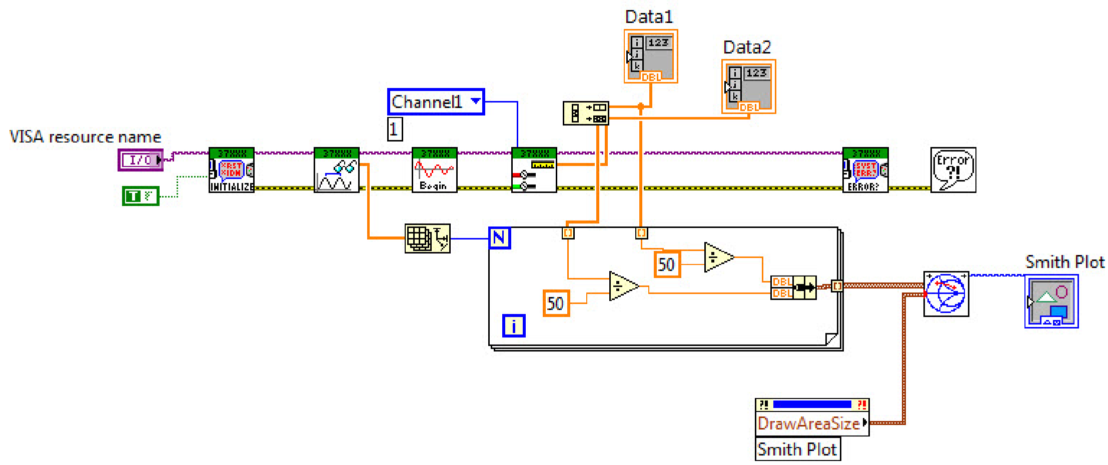


Figure D-56. VI Block Diagram

Below is the Front Panel Display where the data is from a simulated MS464xB.

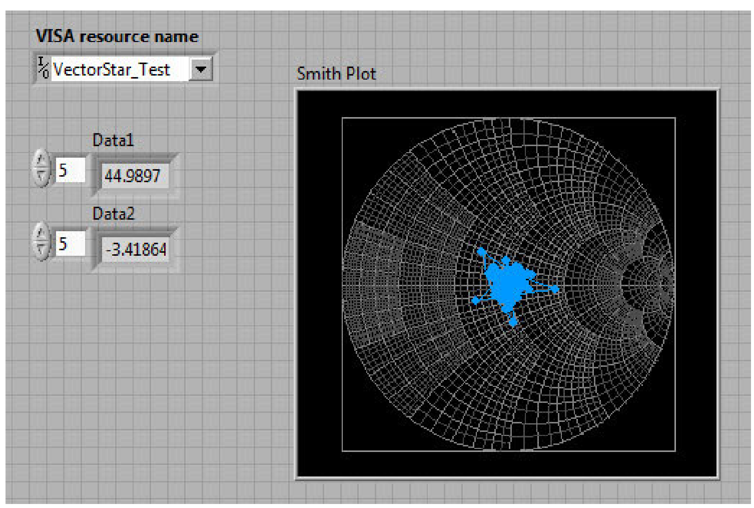


Figure D-57. Front Panel for Example 5

Below is the equivalent display on the MS464xB.

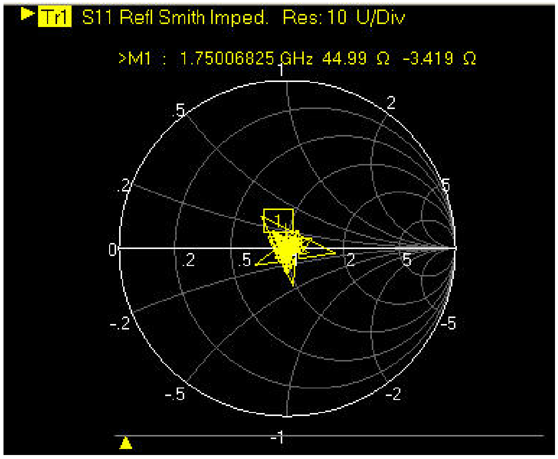


Figure D-58. MS464xB VNA Smith Chart Display

Example 6 – Output S2P File

This example uses VISA Writes and 37XXX driver VIs to accomplish tasks that the driver can't perform. Specifically, sending Native MS464xB commands with Lightning commands to output an S2P file from the VNA to the PC. Since the driver VI is saved as a Private VI in the version 1.1 driver it must be made public (as shown in Figure D-60).

A combination of VISA and driver commands is used to first send Native GPIB strings to setup the S2P output to 25 data points, Frequency in Hz, and Data Format to Real/Imaginary via the Read ASCII ARB function (the private driver to be made public, and then included directly to the VI).

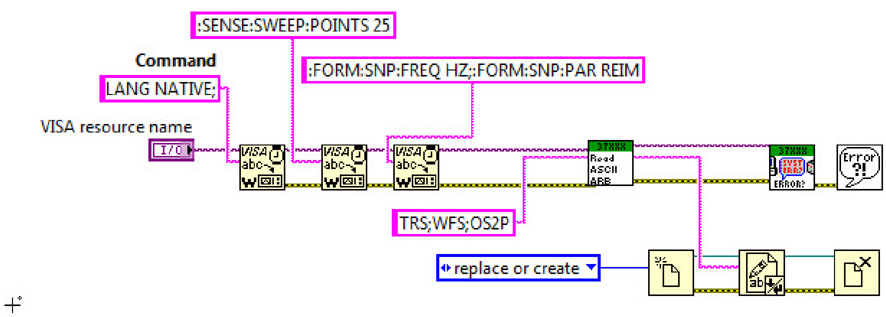


Figure D-59. VI Block Diagram

Several of the Private functions in the driver are useful and can be made public by using Access Scope in the Project Explorer.

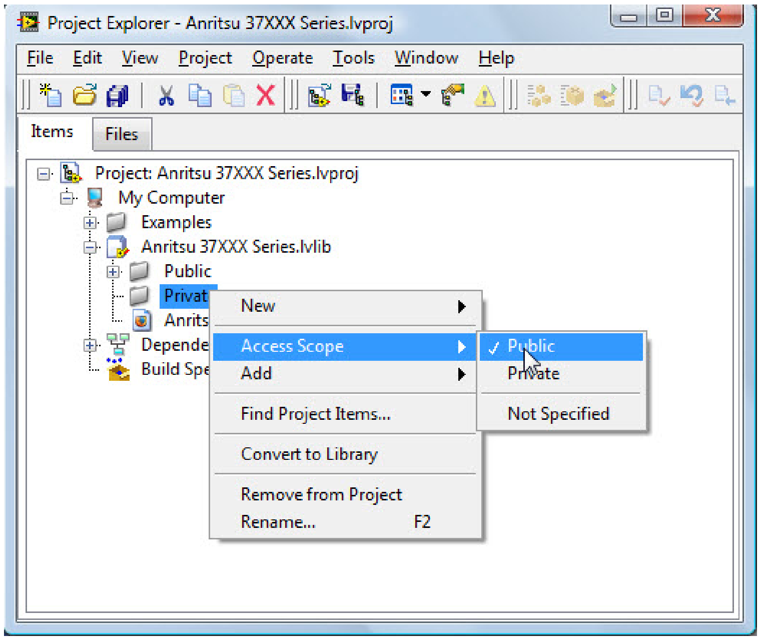
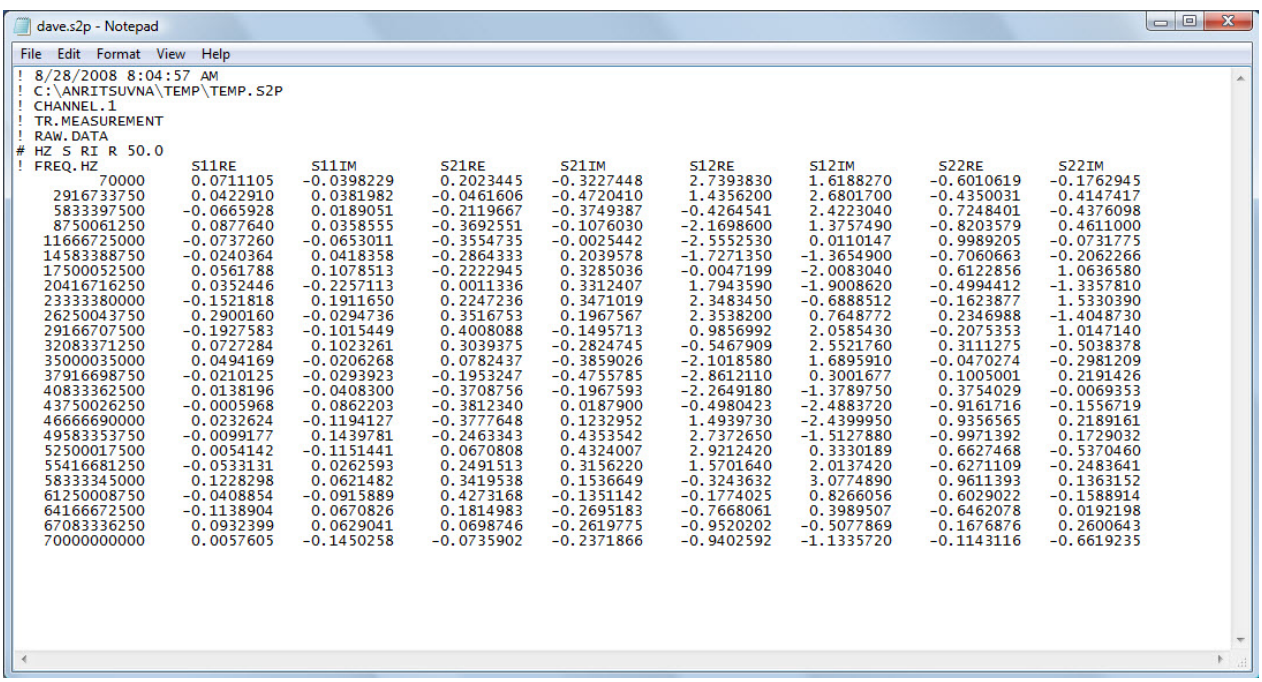


Figure D-60. Project Explorer

When the VI runs, a dialog allows the transferred S2P file name to be selected. Make sure to save the file with an “.S2P” extension.




```
! 8/28/2008 8:04:57 AM
! C:\ANRITSUVNA\TEMP\TEMP.S2P
! CHANNEL.1
! TR.MEASUREMENT
! RAW.DATA
# HZ S RI R 50.0
! FREQ.HZ       S11RE      S11IM      S21RE      S21IM      S12RE      S12IM      S22RE      S22IM
      70000  0.0711105 -0.0398229  0.2023445 -0.3227448  2.7393830  1.6188270 -0.6010619 -0.1762945
 2916733750  0.0422910  0.0381982 -0.0461606 -0.4720410  1.4356200  2.6801700 -0.4350031  0.4147417
 5833397500 -0.0665928  0.0189051 -0.2119667 -0.3749387 -0.4264541  2.4223040  0.7248401 -0.4376098
 8750061250  0.0877640  0.0358555 -0.3692551 -0.1076030 -2.1698600  1.3757490 -0.8203579  0.4611000
11666725000 -0.0737260 -0.0653011 -0.3554735 -0.0025442 -2.5552530  0.0110147  0.9989205 -0.0731775
14583388750 -0.0240364  0.0418358 -0.2864333  0.2039578 -1.7271350 -1.3654900 -0.7060663 -0.2062266
17500052500  0.0561788  0.1078513 -0.2222945  0.3285036 -0.0047199 -2.0083040  0.6122856  1.0636580
20416716250  0.0352446 -0.2257113  0.0011336  0.3312407  1.7943590 -1.9008620 -0.4994412 -1.3357810
23333380000 -0.1521818  0.1911650  0.2247236  0.3471019  2.3483450 -0.6888512 -0.1623877  1.5330390
26250043750  0.2900160 -0.0294736  0.3516753  0.1967567  2.3538200  0.7648772  0.2346988 -1.4048730
29166707500 -0.1927583 -0.1015449  0.4008088 -0.1495713  0.9856992  2.0585430 -0.2075353  1.0147140
32083371250  0.0727284  0.1023261  0.3039375 -0.2824745 -0.5467909  2.5521760  0.3111275 -0.5038378
35000035000  0.0494169 -0.0206268  0.0782437 -0.3859026 -2.1018580  1.6895910 -0.0470274 -0.2981209
37916698750 -0.0210125 -0.0293923 -0.1953247 -0.4755785 -2.8612110  0.3001677  0.1005001  0.2191426
40833362500  0.0138196 -0.0408300 -0.3708756 -0.1967593 -2.2649180 -1.3789750  0.3754029 -0.0069353
43750026250 -0.0005968  0.0862203 -0.3812340  0.0187900 -0.4980423 -2.4883720 -0.9161716 -0.1556719
46666690000  0.0232624 -0.1194127 -0.3777648  0.1232952  1.4939730 -2.4399950  0.9356565  0.2189161
49583353750 -0.0099177  0.1439781 -0.2463343  0.4353542  2.7372650 -1.5127880 -0.9971392  0.1729032
52500017500  0.0054142 -0.1151441  0.0670808  0.4324007  2.9212420  0.3330189  0.6627468 -0.5370460
55416681250 -0.0533131  0.0262593  0.2491513  0.3156220  1.5701640  2.0137420 -0.6271109 -0.2483641
58333345000  0.1228298  0.0621482  0.3419538  0.1536649 -0.3243632  3.0774890  0.9611393  0.1363152
61250008750 -0.0408854 -0.0915889  0.4273168 -0.1351142 -0.1774025  0.8266056  0.6029022 -0.1588914
64166672500 -0.1138904  0.0670826  0.1814983 -0.2695183 -0.7668061  0.3989507 -0.6462078  0.0192198
67083336250  0.0932399  0.0629041  0.0698746 -0.2619775 -0.9520202 -0.5077869  0.1676876  0.2600643
70000000000  0.0057605 -0.1450258 -0.0735902 -0.2371866 -0.9402592 -1.1335720 -0.1143116 -0.6619235
```

Figure D-61. S2P File

Example 7 – Output BMP File

A similar technique as that used in example 6 gets the bitmap data to a file. The Lightning commands “BMPC;OBMP” outputs a bitmap file. BMPC selects color on white as the color scheme, making for better printouts. The “Read ASCII ARB.vi” is used to strip off the arbitrary block header and place the bitmap data into a file (the header corrupts the bitmap file).

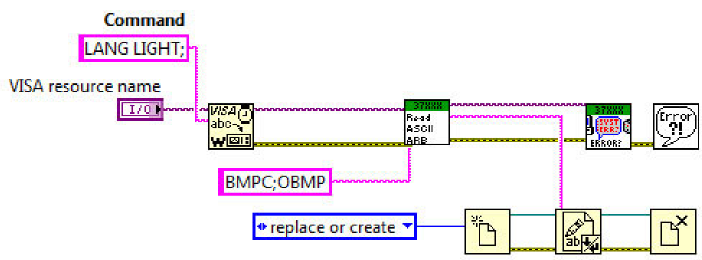


Figure D-62. VI Block Diagram

When the VI runs, a dialog allows the file name to be selected. Make sure to save the file with a “.bmp” extension.

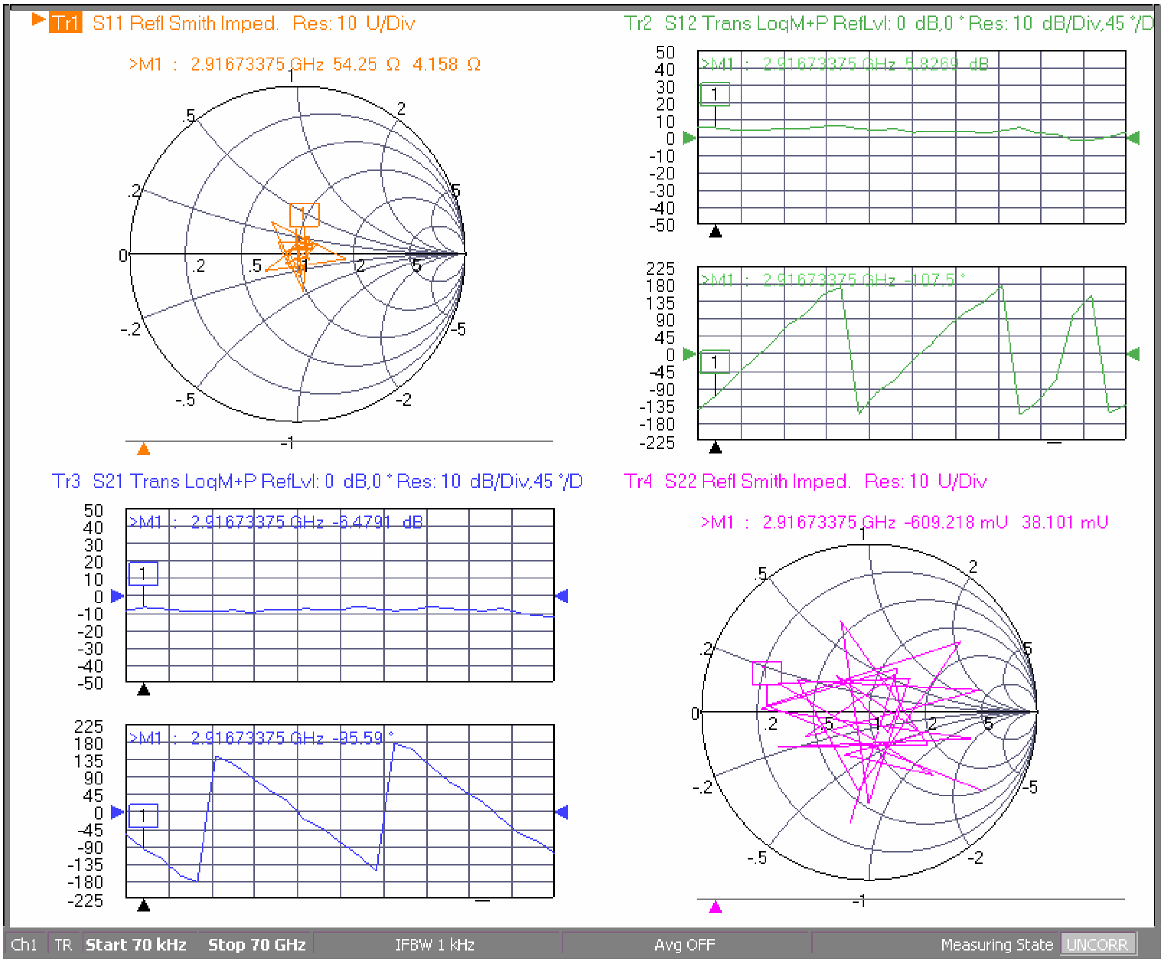


Figure D-63. MS464xB VNA Bitmap Display

D-8 LabWindows/CVI Programming Examples

The examples in this section demonstrate the use of the au37xxx CVI driver for controlling the MS464xB. The following few steps are necessary before getting started with programming.

1. Set up communications to the MS464xB using VXI-11 (TCP/IP) by noting the IP Address of the VNA and set up a resource (a connection string).

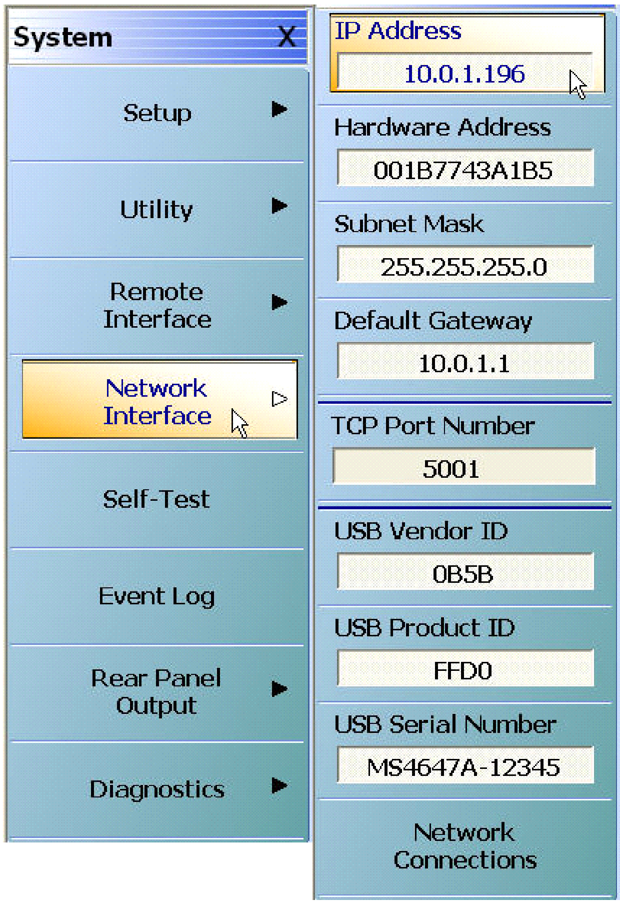


Figure D-64. MS464xB IP Address

2. Launch NI-MAX and create a new resource. The programs will reference this resource rather than a specific address.

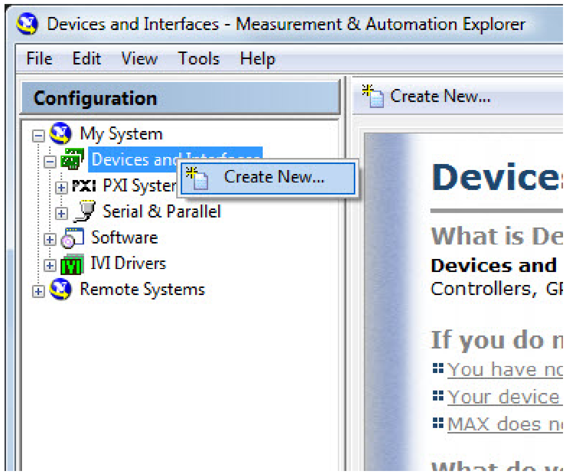


Figure D-65. Measurement and Automation Explorer (MAX)

3. Select the VISA TCP/IP Resource.

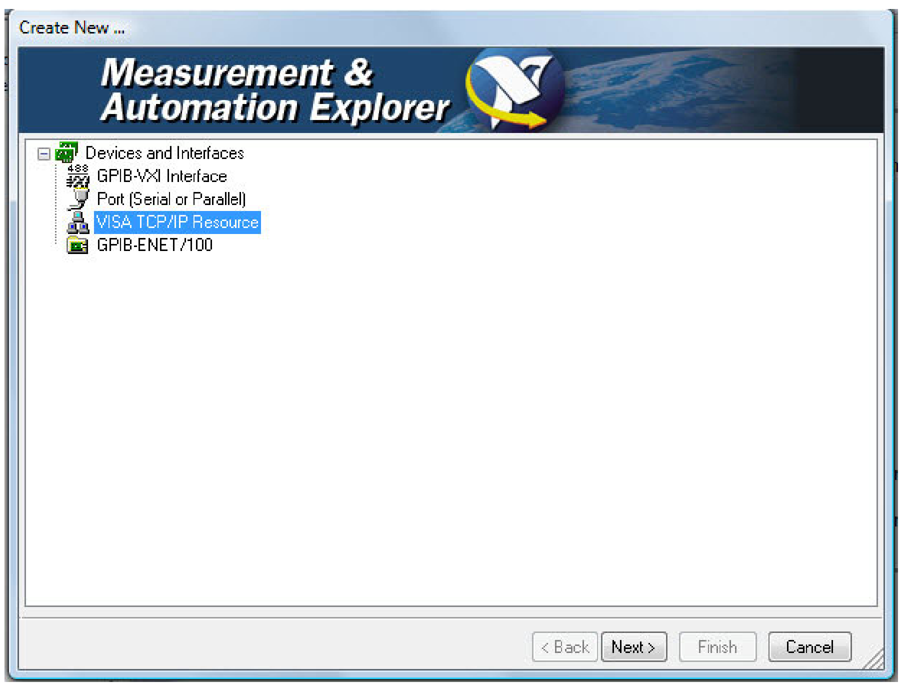


Figure D-66. Creating a New Resource Dialog

4. If the controlling PC and the VNA are on the same local sub-network (this is usually true if the first 3 numbers in the IP address are the same – for example, 10.0.1.x in this case), then you can Auto-detect the VNA. Otherwise, you need to manually enter the IP address.

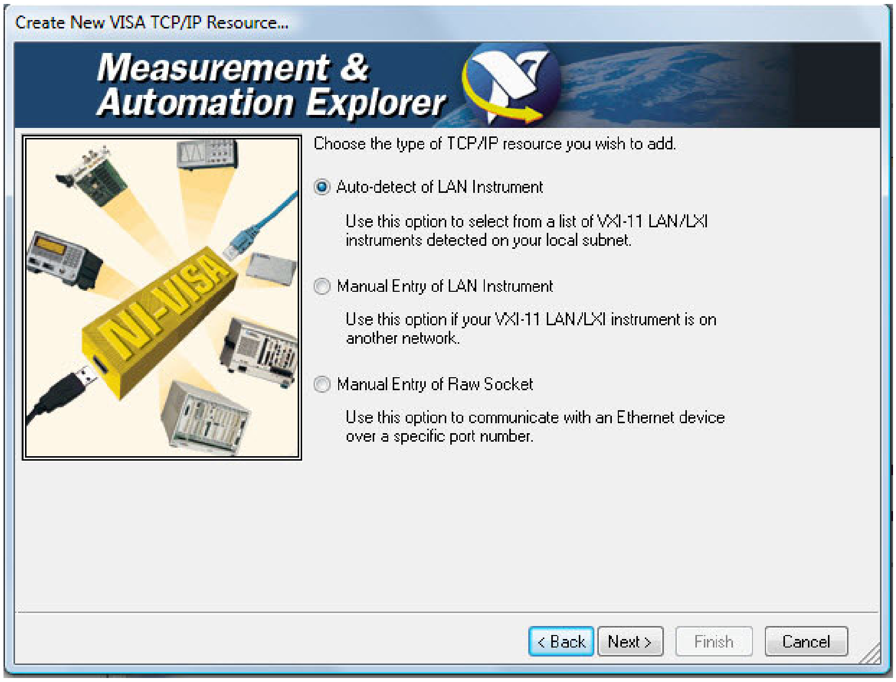


Figure D-67. Create New VISA TCP/IP Resource Dialog

5. Select the detected instrument.

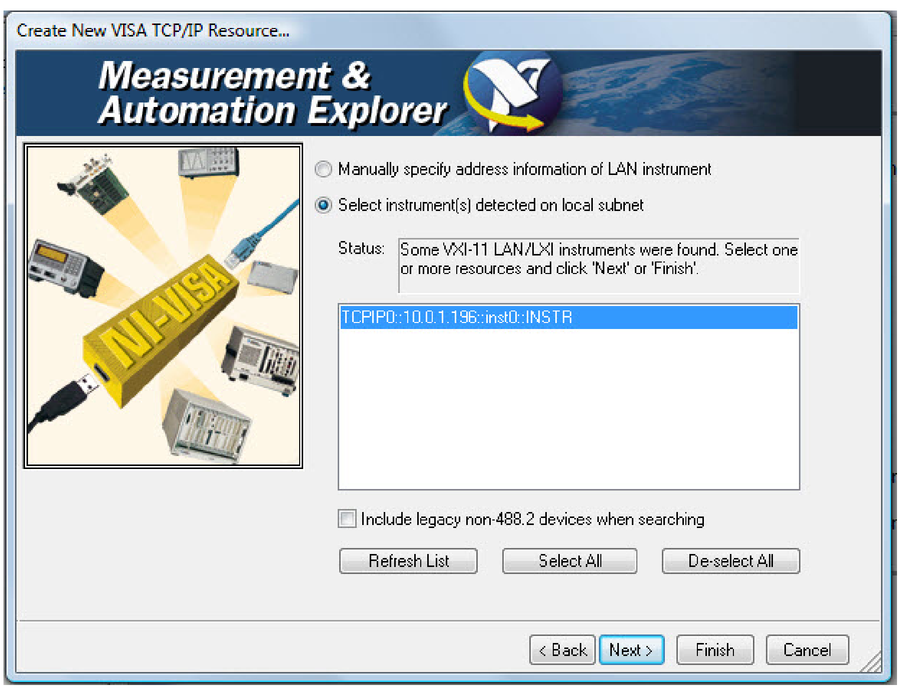


Figure D-68. Create New VISA TCP/IP Resource Dialog

6. Give the instrument an alias (the VectorStar_Test alias is used in LabWindows/CVI).

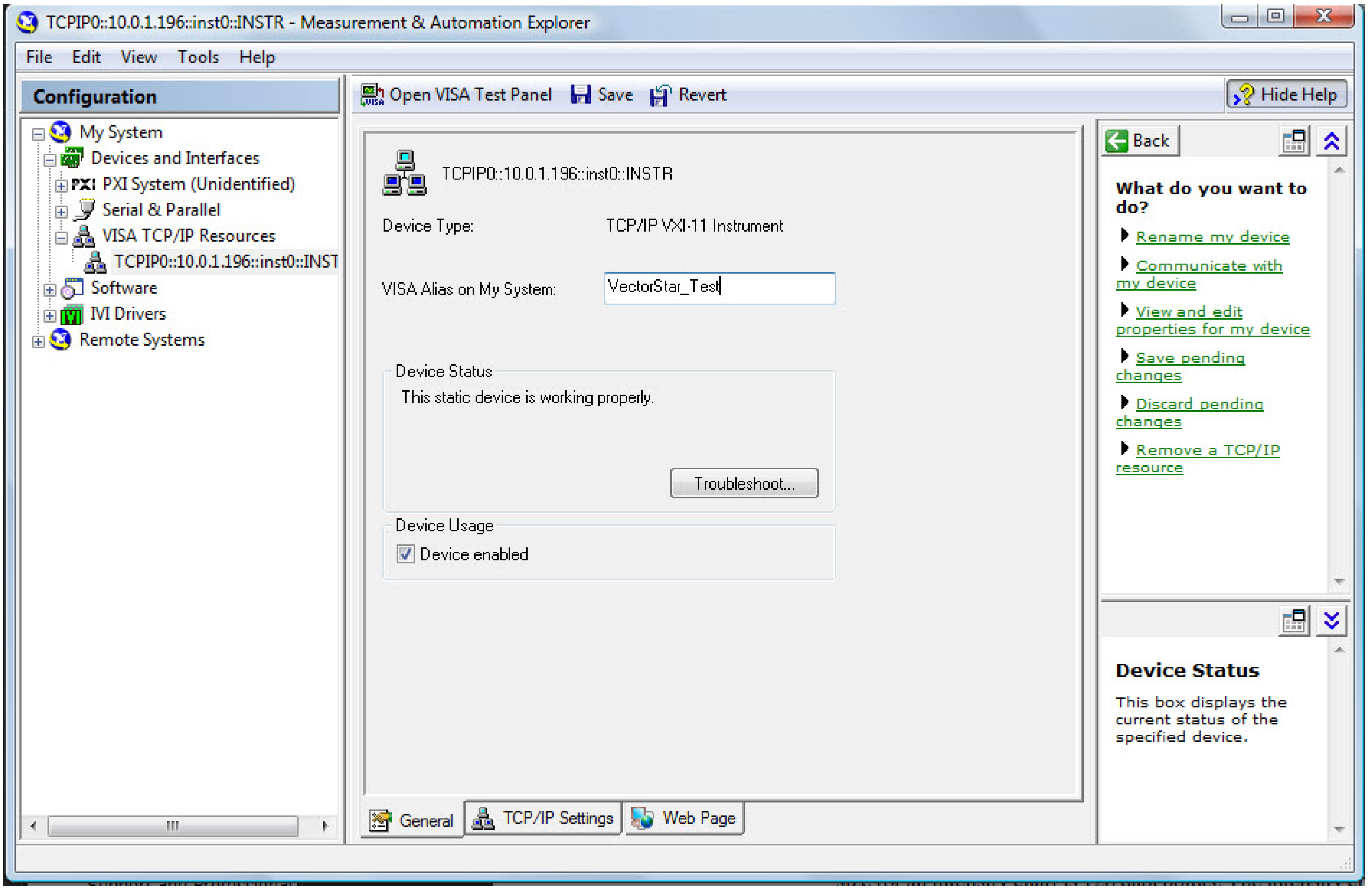


Figure D-69. Create New VISA TCP/IP Resource Dialog

7. Note that the VISA connection string has been replaced with the VISA alias.

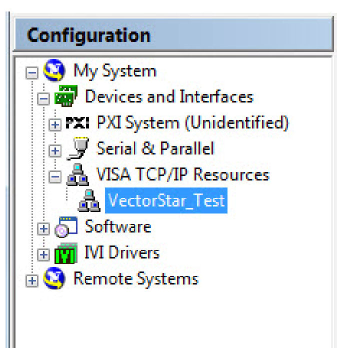


Figure D-70. Measurement and Automation Explorer Configuration

8. In LabWindows/CVI, create a new Project from Template.

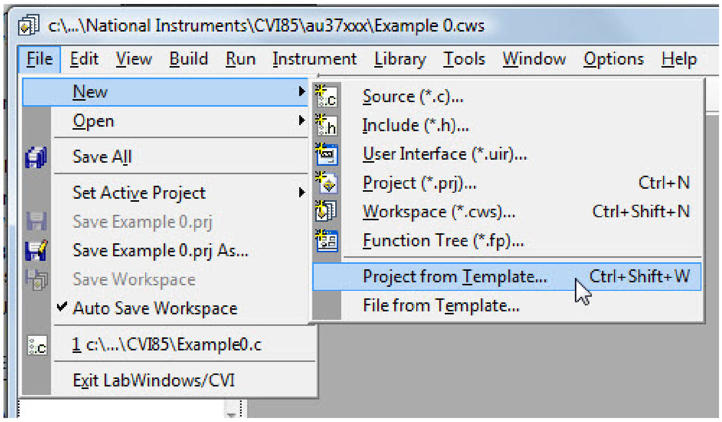


Figure D-71.

9. Use the Command-line Application Template.

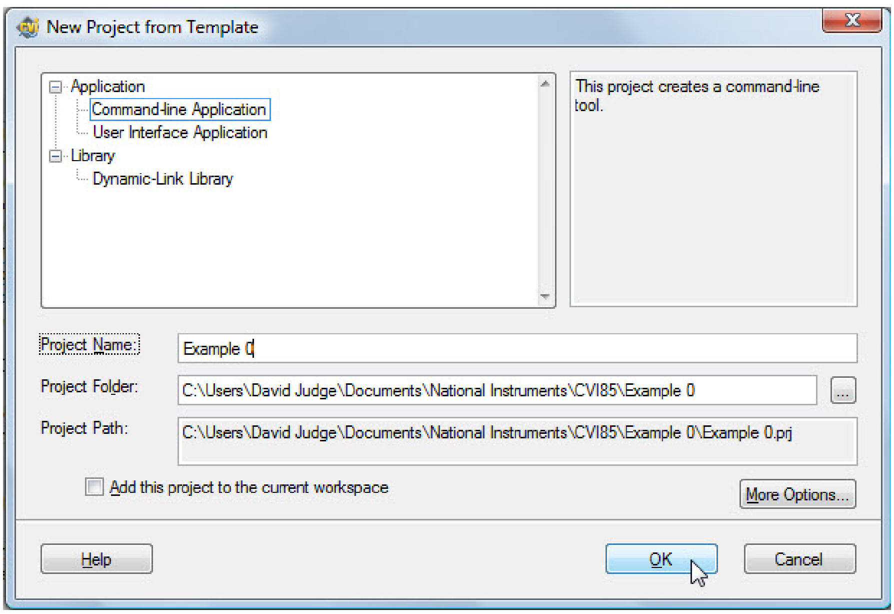


Figure D-72.

10. Set up the CVI to load the Anritsu driver automatically every time CVI starts up in: Library | Customize to have CVI load the driver into our user Library every time CVI starts. This is a preferable way to have access to the driver.

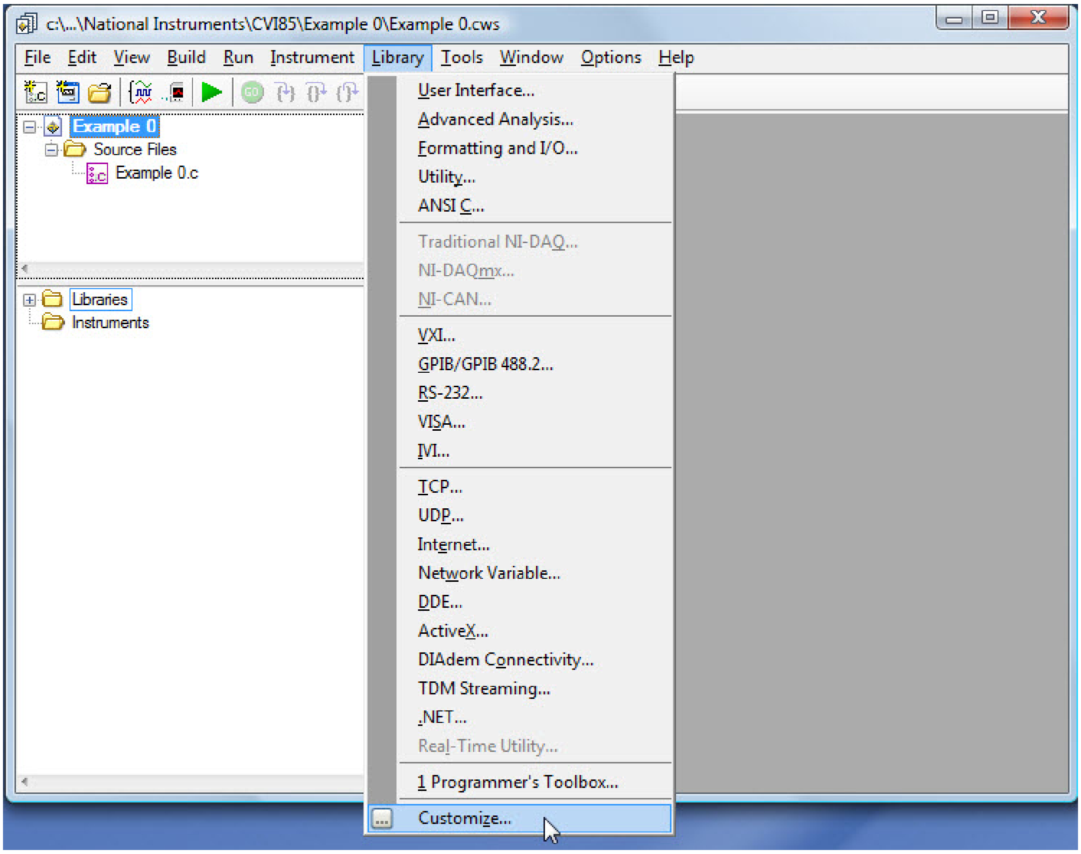


Figure D-73. Customizing CVI

11. Browse to find the au37xxx.fp file.

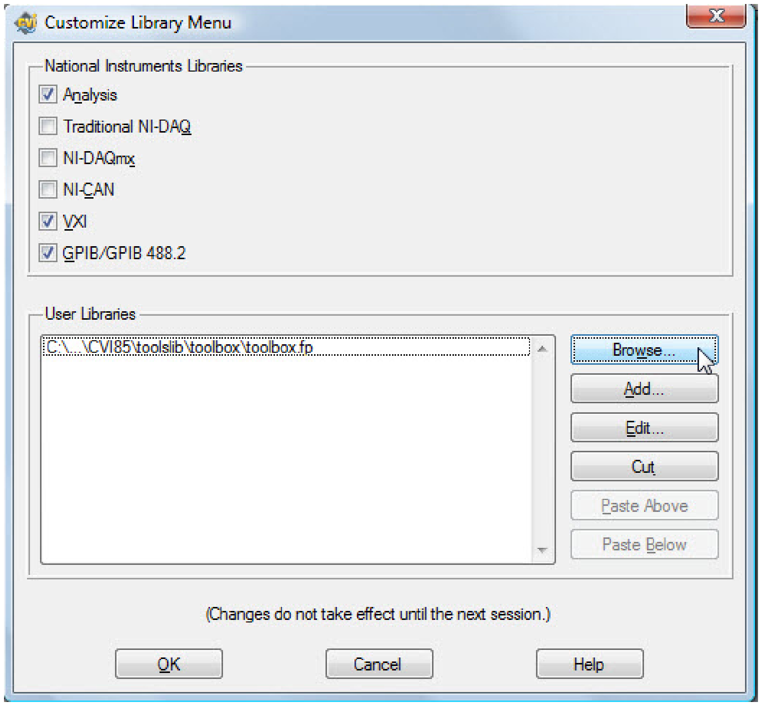


Figure D-74. Customize Library Menu

12. Add the Function Tree (".fp") file.

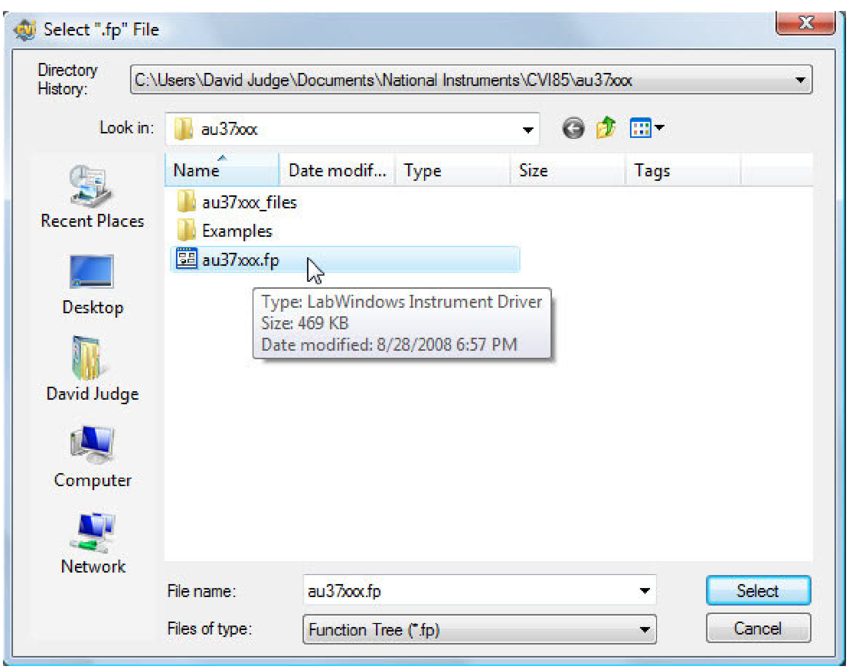


Figure D-75. Selecting File

13. The driver is now in the Libraries folder.

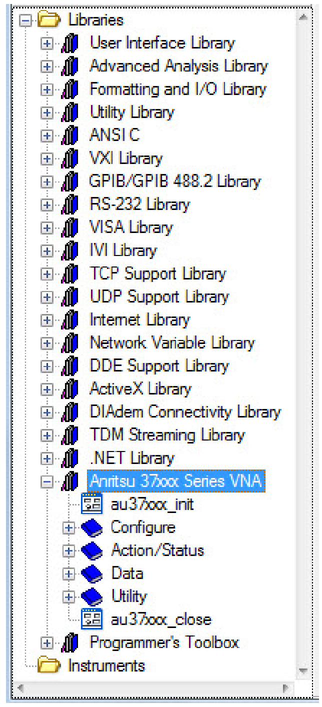


Figure D-76. Driver Library

14. Copy the driver DLL to someplace on the on the system path. If the system path is not known or a custom path is not important, copy the file to C:\Windows\System32.

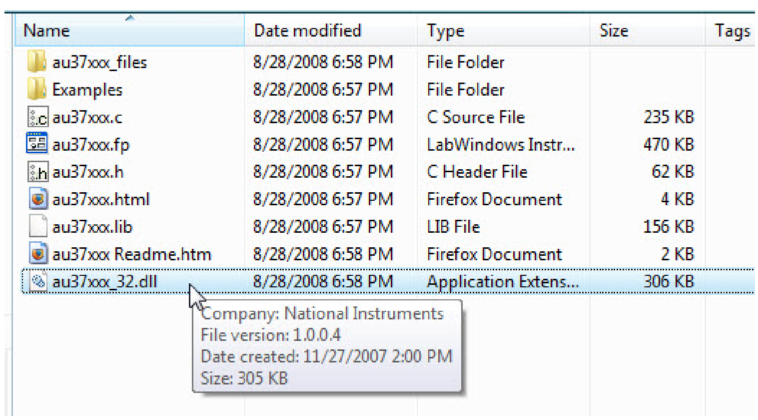


Figure D-77. Copying Driver DLL

15. Copy the DLL to C:\Windows\System32. This step is necessary (the LabWindows/CVI documentation explains more about copying the DLL file).

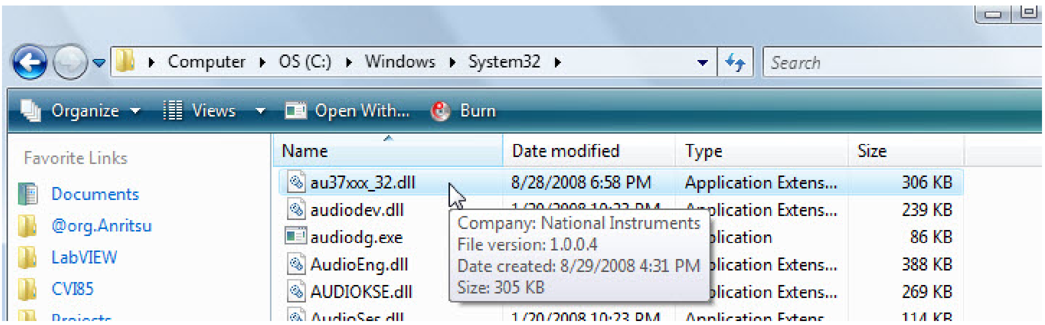


Figure D-78. Copying Driver DLL

Example 0 – Opening a Session

This first example opens a communication session to the VNA and then uses two of the driver VIs to get some information about the VNA.

```
// Include files
#include <ansi_c.h>
#include "au37xxx.h"

int main (int argc, char *argv[])
{
    ViSession session;
    ViStatus status;
    ViChar d[256];
    ViChar d1[256];

    printf("Example 0: Connection and Revision Query\n\n");
    //status = au37xxx_init ("TCPIP0::10.0.1.196::INSTR", VI_TRUE, VI_FALSE,
&session);
    status = au37xxx_init ("VectorStar_Test", VI_TRUE, VI_FALSE, &session);
    status = au37xxx_revision_query (session, d,d1);
    au37xxx_close(session);
    printf("Driver Version: %s\n",d);
    printf("Firmware Version: %s\n",d1);

    printf("\n\nHit return to exit:");
    getc(stdin);

    return 0;
}
```

The results of running Example 0 are shown below. The au37xxx_revision_query() function returns two strings. The first is the version of the driver and the second is the version of firmware on the VNA.

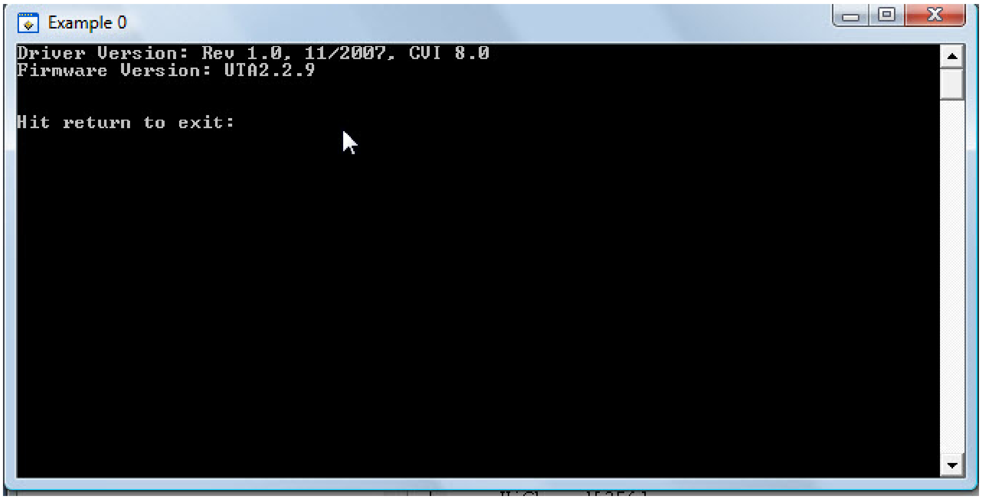


Figure D-79. Example 0

Example 1 – Sending the *IDN? Command and Displaying Results

The previous example used only driver functions to get some information from the VNA. The GPIB command, “*IDN?” returns the Manufacturer, Model #, Serial Number and Firmware Version. This command was used previously in the Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser. In this example, the “*IDN?” command is directly issued and then the different parts of the response string are parsed.

```
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include "au37xxx.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au37xxx_status = checkErr((fCal)), au37xxx_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{

    ViStatus status;
    ViUInt32 read_count;
    ViStatus    au37xxx_status   = VI_SUCCESS;
    ViChar l_buffer[50];
    ViChar* p2Manf = NULL;
    ViChar* p2Model = NULL;
    ViChar* p2Ser = NULL;
    ViChar* p2Firm = NULL;

    printf("Example 1: Using the *IDN? Query\n\n");
    CHECKERR(au37xxx_init ("VectorStar_Test", VI_TRUE, VI_FALSE, &session));
    CHECKERR(au37xxx_write (session, "*IDN?"));
    CHECKERR(viRead (session, (ViPBuf)l_buffer, 50, &read_count));
```

```
    au37xxx_close(session);
    p2Manf = strtok(l_buffer, ",");
    p2Model = strtok(NULL, ",");
    p2Ser = strtok(NULL, ",");
    p2Firm = strtok(NULL, "\n");

    printf("Manufacturer: %s\nModel: %s\nSer#: %s\nFirmware: %s\n",
        p2Manf,p2Model,p2Ser,p2Firm);

    printf("\n\nHit return to exit:");
    getc(stdin);

    return 0;

Error:
    printf("\n\nDetected an Error--Hit return to exit:");
    getc(stdin);

    return 0;
}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];

    if (status < VI_SUCCESS)
    {
        au37xxx_error_message (session, status, error_message);
        sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status,
error_message);
        MessagePopup ("Error", error_buffer);
        au37xxx_error_query (session, error_message);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        MessagePopup ("Error", error_buffer);
        au37xxx_close(session);
        session = 0;
    }
    return status;
}
```

The output from Example 1 is shown below. The au37xxx_write() function is used to directly send GPIB commands. The VISA function viRead() is then used to read the results.

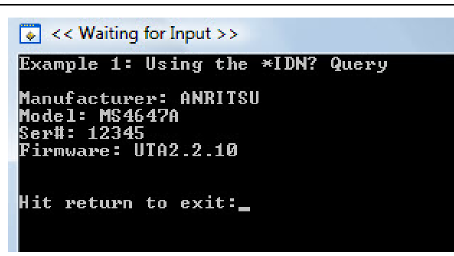


Figure D-80. Example 1

Example 2 – Error Checking

This example shows that if an invalid GPIB string is sent to the VNA then the CHECKERR macro catches the error and displays the error message from the VNA. Here we send two valid strings: "*IDN?" and then "OID". Note that the third string is not a valid GPIB command and the instrument reports this.

```
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include "au37xxx.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au37xxx_status = checkErr((fCal)), au37xxx_status < VI_SUCCESS) \
        goto Error; else


int main (int argc, char *argv[])
{

    ViStatus status;
    ViUInt32 read_count;
    ViStatus au37xxx_status = VI_SUCCESS;
    ViChar l_buffer[50];
    ViChar* p2Manf = NULL;
    ViChar* p2Model = NULL;
    ViChar* p2Ser = NULL;
    ViChar* p2Firm = NULL;

    printf("Example 2: Testing for Errors\n\n");
    CHECKERR(au37xxx_init ("VectorStar_Test", VI_TRUE, VI_FALSE, &session));

    //Use Native VNA Error Checking
    CHECKERR(au37xxx_write (session, "LANG NATIVE"));

    //First send a known good command
    CHECKERR(au37xxx_write (session, "*IDN?"));
    CHECKERR(viRead (session, (ViPBuf)l_buffer, 50, &read_count));

    //Send an OID - also a good command
    CHECKERR(au37xxx_write (session, "OID"));
    CHECKERR(viRead (session, (ViPBuf)l_buffer, 50, &read_count));

    //This command is not a valid VNA command and should generate an error
    CHECKERR(au37xxx_write (session, "ABC"));
    au37xxx_close(session);

    printf("\n\nHit return to exit:");
    getc(stdin);

    return 0;

Error:
    printf("\n\nDetected an Error--Hit return to exit:");
    getc(stdin);
```

```
    return 0;
}


ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4;

    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set

    if ((status < VI_SUCCESS) | |  ((stb & VNA_ERROR) > 0))
    {
        au37xxx_error_message (session, status, error_message);
        sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status,
error_message);
        printf ("%s\n", error_buffer);
        au37xxx_error_query (session, error_message);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
        au37xxx_close(session);
        session = 0;
    }
    return status;
}
```

The Service Request Status Register is shown below and is slightly changed from Lightning to the MS464xB VNA. If LANG LIGHT is set, then the Lightning configuration of the Status Register is used. In this example the “LANG NATIVE” command is sent to use the MS464xB status register. The code checks b2 to see if the error queue is not empty.

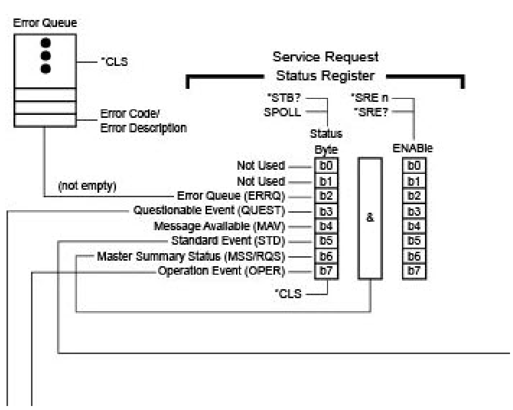


Figure D-81. MS464xB Service Request Status Register

The command error is caught after sending the erroneous “ABC” command and reports the message, “Faulty program mnemonic syntax.”

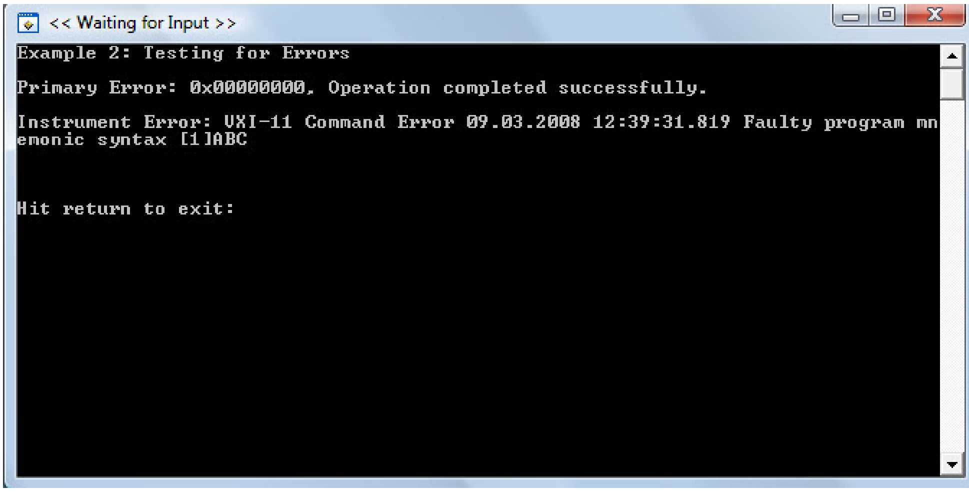


Figure D-82. Example 2

Example 3 – Sending Data to a File with the LIST Command

The Lightning commands “FMT1;LIST” are sent to get the full list of commands supported by the MS464xB. The sting returned is an ASCII arbitrary block and the au37xxx_readAsciiARBBlock() function is used to strip off the arbitrary block header. The results are then sent to a file.

```
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au37xxx.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au37xxx_status = checkErr((fCal)), au37xxx_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{

    ViInt32 retCount;

    ViStatus status;
    ViUInt32 read_count;
    ViStatus au37xxx_status = VI_SUCCESS;
    int fileHandle;
    static ViChar readBuffer[100000];

    CHECKERR(au37xxx_init ("VectorStar_Test", VI_FALSE, VI_FALSE, &session));
    CHECKERR(au37xxx_write(session,"FMT1;LIST"));

    CHECKERR(au37xxx_readAsciiARBBlock(session,100000,readBuffer,&retCount));
    au37xxx_close(session);
```

```
    fileHandle = OpenFile (".\\commands.txt", VAL_WRITE_ONLY, VAL_OPEN_AS_IS,
VAL_ASCII);
    WriteFile (fileHandle, readBuffer, retCount);
    CloseFile (fileHandle);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4;

    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set

    if ((status < VI_SUCCESS) ||  ((stb & VNA_ERROR) > 0))
    {
        au37xxx_error_message (session, status, error_message);
        sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status,
error_message);
        printf ("%s\n", error_buffer);
        au37xxx_error_query (session, error_message);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
        au37xxx_close(session);
        session = 0;
    }
    return status;
}
```

The list of all commands supported by the MS464xB are listed in the commands.txt file.

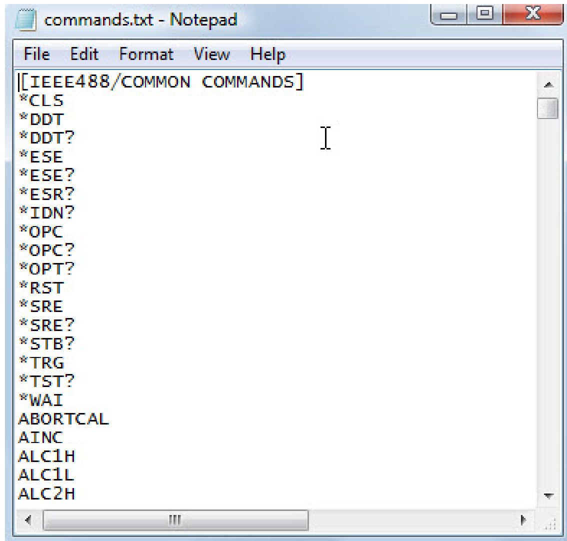


Figure D-83. MS464xB VNA Programming Commands List

Use the Anritsu GPIB, USB, VXI-11 Exerciser to get more help on any command. Help will tell you what type of command (Native, Lightning, HP8510) and provides syntax.

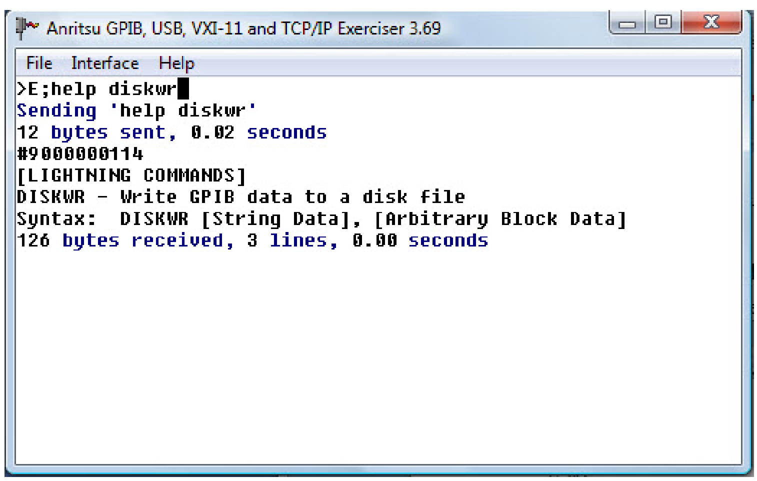


Figure D-84. Anritsu GPIB, USB, VXI-11, and TCP/IP Exerciser

Example 4 – Acquiring Trace Data

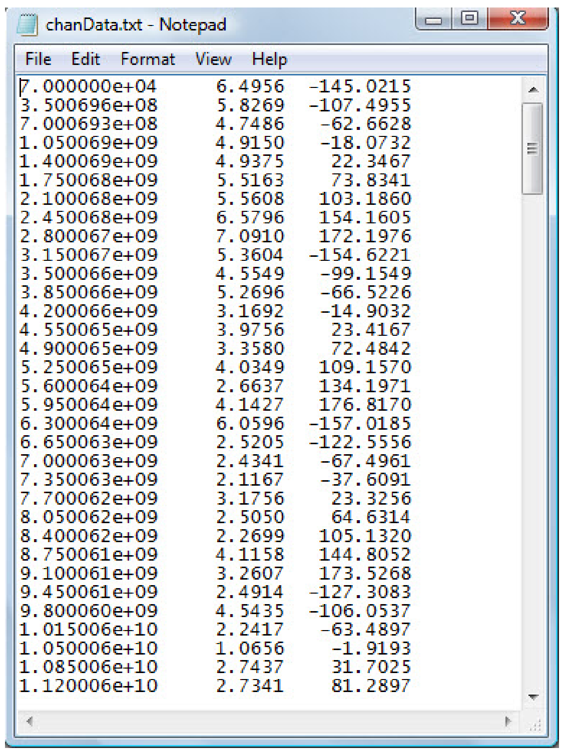
In this example, the final data from Trace 2, which is set to Log magnitude and Phase data, is acquired programmatically. The data comes out in a one dimensional, interleaved array. The array must be parsed to get the log magnitude and phase data into two separate arrays.

```
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au37xxx.h"

ViSession session;
ViStatus checkErr (ViStatus status);
```

```
#define CHECKERR(fCal) \
    if (au37xxx_status = checkErr((fCal)), au37xxx_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{

    ViInt32 retCount;
    ViStatus status;
    ViUInt32 read_count;
    ViStatus au37xxx_status = VI_SUCCESS;
    FILE* fp;
    ViChar readBuffer[500000];
    ViReal64 fdata[201];
    ViReal64 chanData[402];
    ViReal64 lmData[201];
    ViReal64 phaseData[201];
    int i, ii = 0;

    CHECKERR(au37xxx_init ("VectorStar_Test", VI_FALSE, VI_FALSE, &session));
    CHECKERR(au37xxx_configureChannel (session, 2));
    CHECKERR(au37xxx_configureSweep (session, 201, AU37XXX_SWEEP_NORMAL));
    CHECKERR(au37xxx_readFrequencyValues (session, 201,fdata,&retCount));
    CHECKERR(au37xxx_readChannelData (session, 2, AU37XXX_DATA_FINAL, 402,
                  chanData, &retCount));

    au37xxx_close(session);

    fp = fopen(".\\chanData.txt","w") ;

    for (i=0;i<201;i++)
    {
        lmData[i] = chanData[ii++];
        phaseData[i] = chanData[ii++];
        fprintf(fp,"%e %10.4f\t%10.4f\n",fdata[i],lmData[i],phaseData[i]);
    }

    fclose(fp);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4;

    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set
```

```
        if ((status < VI_SUCCESS) ||  ((stb & VNA_ERROR) > 0))
        {
            au37xxx_error_message (session, status, error_message);
            sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status,
  error_message);
            printf ("%s\n", error_buffer);
            au37xxx_error_query (session, error_message);
            SetWaitCursor (0);
            sprintf (error_buffer, "Instrument Error: %s\n", error_message);
            printf ("%s\n", error_buffer);
            au37xxx_close(session);
            session = 0;
        }
        return status;
    }
```

The data is printed out into a three-column format: Frequency (Hz), Log Magnitude, and Phase (note that the data is simulated).



7.000000e+04	6.4956	-145.0215
3.500696e+08	5.8269	-107.4955
7.000693e+08	4.7486	-62.6628
1.050069e+09	4.9150	-18.0732
1.400069e+09	4.9375	22.3467
1.750068e+09	5.5163	73.8341
2.100068e+09	5.5608	103.1860
2.450068e+09	6.5796	154.1605
2.800067e+09	7.0910	172.1976
3.150067e+09	5.3604	-154.6221
3.500066e+09	4.5549	-99.1549
3.850066e+09	5.2696	-66.5226
4.200066e+09	3.1692	-14.9032
4.550065e+09	3.9756	23.4167
4.900065e+09	3.3580	72.4842
5.250065e+09	4.0349	109.1570
5.600064e+09	2.6637	134.1971
5.950064e+09	4.1427	176.8170
6.300064e+09	6.0596	-157.0185
6.650063e+09	2.5205	-122.5556
7.000063e+09	2.4341	-67.4961
7.350063e+09	2.1167	-37.6091
7.700062e+09	3.1756	23.3256
8.050062e+09	2.5050	64.6314
8.400062e+09	2.2699	105.1320
8.750061e+09	4.1158	144.8052
9.100061e+09	3.2607	173.5268
9.450061e+09	2.4914	-127.3083
9.800060e+09	4.5435	-106.0537
1.015006e+10	2.2417	-63.4897
1.050006e+10	1.0656	-1.9193
1.085006e+10	2.7437	31.7025
1.120006e+10	2.7341	81.2897

Figure D-85. Simulated Trace Data

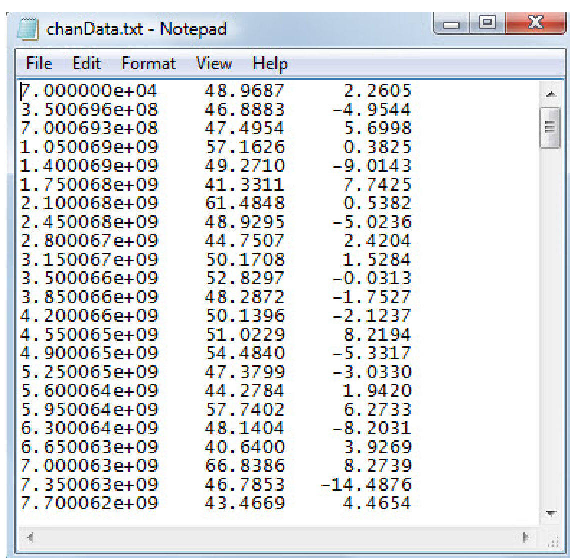
Example 5 – Smith Chart Data

This example is similar to the previous example, except the instrument is **not** reset, which sets up smith charts on trace 1 and 4.

```
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au37xxx.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au37xxx_status = checkErr((fCal)), au37xxx_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{
    ViInt32 retCount;
    ViStatus status;
    ViUInt32 read_count;
    ViStatus au37xxx_status = VI_SUCCESS;
    FILE* fp;
    ViChar readBuffer[500000];
    ViReal64 fdata[201];
    ViReal64 chanData[402];
    ViReal64 impData[201];
    ViReal64 reactData[201];
    int i, ii = 0;

    CHECKERR(au37xxx_init ("VectorStar_Test", VI_FALSE, VI_TRUE, &session));
    CHECKERR(au37xxx_configureChannel (session, 1));
    CHECKERR(au37xxx_configureSweep (session, 201, AU37XXX_SWEEP_NORMAL));
    CHECKERR(au37xxx_readFrequencyValues (session, 201,fdata,&retCount));
    CHECKERR(au37xxx_readChannelData (session, 1, AU37XXX_DATA_FINAL, 402,
             chanData, &retCount));

    au37xxx_close(session);

    fp = fopen(".\\chanData.txt","w") ;
    for (i=0;i<201;i++)
    {
        impData[i] = chanData[ii++];
        reactData[i] = chanData[ii++];
        fprintf(fp,"%e %10.4f\t%10.4f\n",fdata[i],impData[i],reactData[i]);
    }

    fclose(fp);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}
```

```
ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4;

    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set

    if ((status < VI_SUCCESS) ||  ((stb & VNA_ERROR) > 0))
    {
        au37xxx_error_message (session, status, error_message);
        sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status,
error_message);
        printf ("%s\n", error_buffer);
        au37xxx_error_query (session, error_message);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
        au37xxx_close(session);
        session = 0;
    }
    return status;
}
```

The impedance/reactance data from Trace 1 (Smith Chart) is parsed into an ASCII file with a three-column format: Frequency (Hz), Impedance, and Reactance (note that the data is simulated).



7.000000e+04	48.9687	2.2605
3.500696e+08	46.8883	-4.9544
7.000693e+08	47.4954	5.6998
1.050069e+09	57.1626	0.3825
1.400069e+09	49.2710	-9.0143
1.750068e+09	41.3311	7.7425
2.100068e+09	61.4848	0.5382
2.450068e+09	48.9295	-5.0236
2.800067e+09	44.7507	2.4204
3.150067e+09	50.1708	1.5284
3.500066e+09	52.8297	-0.0313
3.850066e+09	48.2872	-1.7527
4.200066e+09	50.1396	-2.1237
4.550065e+09	51.0229	8.2194
4.900065e+09	54.4840	-5.3317
5.250065e+09	47.3799	-3.0330
5.600064e+09	44.2784	1.9420
5.950064e+09	57.7402	6.2733
6.300064e+09	48.1404	-8.2031
6.650063e+09	40.6400	3.9269
7.000063e+09	66.8386	8.2739
7.350063e+09	46.7853	-14.4876
7.700062e+09	43.4669	4.4654

Figure D-86. Simulated Trace Data

Example 5 is modified to use the built-in Smith chart control. The following program is adapted from the Smith Chart Demo in the samples\apps\smithchart directory. The smith chart fp, found in toolslib\toolbox\smith.fp. is used. By default, Trace1 is set to output impedance values. Most Smith chart controls actually take a normalized impedance (normalized to 1), so the impedance/reactance pairs are divided by 50 ohms to get normalized smith chart data.

```
ViInt32 retCount;
ViStatus status;
ViUInt32 read_count;
ViStatus au37xxx_status = VI_SUCCESS;

ViReal64 fdata[201];
ViReal64 chanData[402];
ViReal64 impData[201];
ViReal64 reactData[201];
int i, ii = 0;

CHECKERR(au37xxx_init ("VectorStar_Test", VI_FALSE, VI_FALSE, &session));
CHECKERR(au37xxx_configureChannel (session, 1));
CHECKERR(au37xxx_configureSweep (session, 201, AU37XXX_SWEEP_NORMAL));
CHECKERR(au37xxx_readFrequencyValues (session, 201,fdata,&retCount));
CHECKERR(au37xxx_readChannelData (session, 1, AU37XXX_DATA_FINAL, 402,
   chanData, &retCount));

au37xxx_close(session);

for (i=0;i<201;i++)
{
    impData[i] = chanData[ii++];
    reactData[i] = chanData[ii++];
    gZ.Real = impData[i]/50.0;
    gZ.Im = reactData[i]/50.0;

    SMITH_PlotImpedancePoint(panelHandle, PANEL_GRAPH, &gZ,
       VAL_SOLID_CIRCLE, IMPEDANCE_COLOR);
}
```

The Labwindows/CVI smith.fp is used to plot smith chart data.

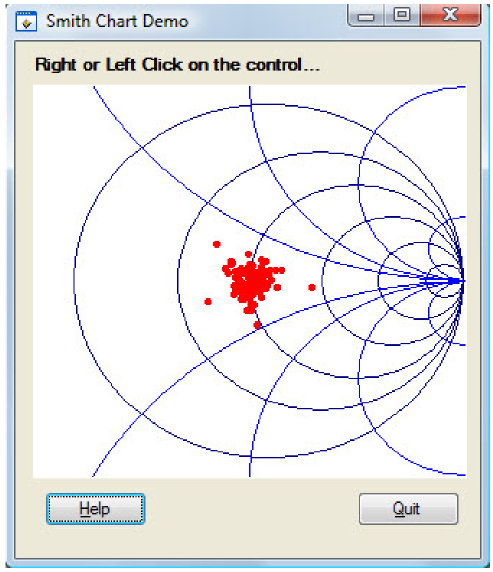


Figure D-87. Smith Chart Demo

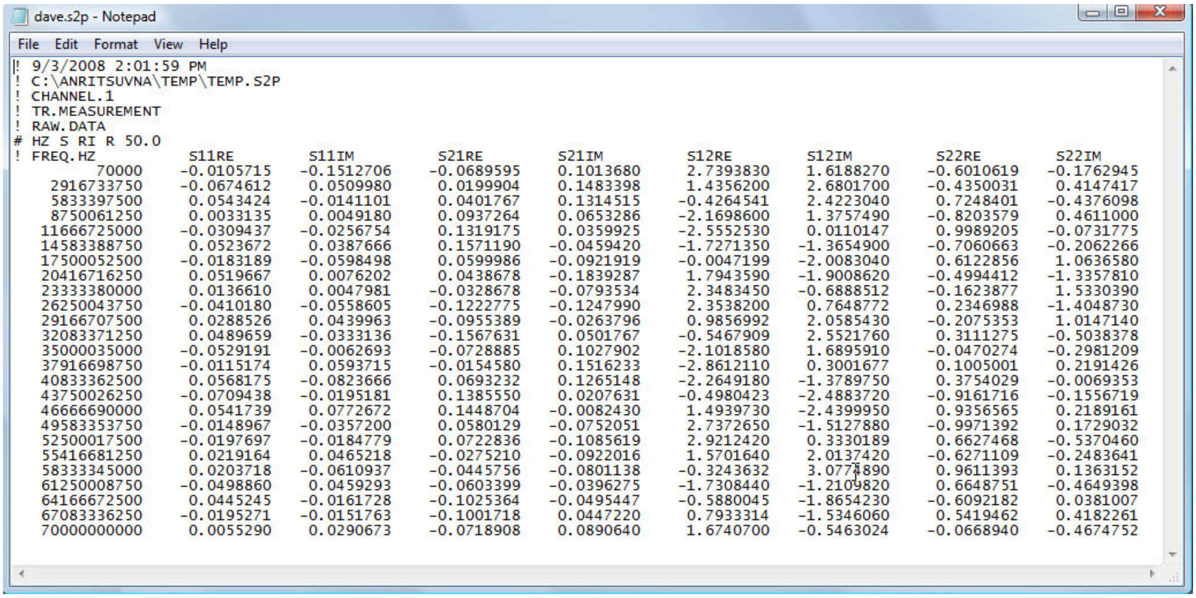
Example 6 – Output S2P File

This example uses au37xxx_write() calls to accomplish tasks that the driver can't perform. Specifically, to send Native MS464xB commands along with Lightning commands to output an S2P file from the VNA to the PC.

```
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au37xxx.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au37xxx_status = checkErr((fCal)), au37xxx_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{

ViInt32 retCount;

    ViStatus status;
    ViUInt32 read_count;
    ViStatus au37xxx_status = VI_SUCCESS;
    int fileHandle;
    ViChar readBuffer[100000];

    CHECKERR(au37xxx_init ("VectorStar_Test", VI_FALSE, VI_TRUE, &session));
    CHECKERR(au37xxx_write(session,"LANG NATIVE"));
    CHECKERR(au37xxx_write(session,":SENSE:SWEEP:POINTS 25"));
    CHECKERR(au37xxx_write(session,":FORM:SNP:FREQ HZ"));
    CHECKERR(au37xxx_write(session,":FORM:SNP:PAR REIM"));
```

```
    CHECKERR(au37xxx_write(session,"TRS;WFS;OS2P"));

    CHECKERR(au37xxx_readAsciiARBBlock(session,100000,readBuffer,&retCount));
    au37xxx_close(session);
    fileHandle = OpenFile (".\\dave.s2p", VAL_WRITE_ONLY, VAL_OPEN_AS_IS,
VAL_ASCII);
    WriteFile (fileHandle, readBuffer, retCount);
    CloseFile (fileHandle);

Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}

ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4;

    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set

    if ((status < VI_SUCCESS) | |  ((stb & VNA_ERROR) > 0))
    {
        au37xxx_error_message (session, status, error_message);
        sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status,
error_message);
        printf ("%s\n", error_buffer);
        au37xxx_error_query (session, error_message);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
        au37xxx_close(session);
        session = 0;
    }
    return status;
}
```

The resulting S2P file is transferred to the PC as shown below.



dave.s2p - Notepad

File Edit Format View Help

```
! 9/3/2008 2:01:59 PM
! C:\ANRITSUVNA\TEMP\TEMP.S2P
! CHANNEL.1
! TR.MEASUREMENT
! RAW.DATA
# HZ S RI R 50.0
! FREQ.HZ        S11RE      S11IM       S21RE      S21IM       S12RE      S12IM       S22RE      S22IM
        70000 -0.0105715 -0.1512706 -0.0689595  0.1013680  2.7393830  1.6188270 -0.6010619 -0.1762945
   2916733750 -0.0674612  0.0509980  0.0199904  0.1483398  1.4356200  2.6801700 -0.4350031  0.4147417
   5833397500  0.0543424 -0.0141101  0.0401767  0.1314515 -0.4264541  2.4223040  0.7248401 -0.4376098
   8750061250  0.0033135  0.0049180  0.0937264  0.0653286 -2.1698600  1.3757490 -0.8203579  0.4611000
  11666725000 -0.0309437 -0.0256754  0.1319175  0.0359925 -2.5552530  0.0110147  0.9989205 -0.0731775
  14583388750  0.0523672  0.0387666  0.1571190 -0.0459420 -1.7271350 -1.3654900 -0.7060663 -0.2062266
  17500052500 -0.0183189 -0.0598498  0.0599986 -0.0921919 -0.0047199 -2.0083040  0.6122856  1.0636580
  20416716250  0.0519667  0.0076202  0.0438678 -0.1839287  1.7943590 -1.9008620 -0.4994412 -1.3357810
  23333380000  0.0136610  0.0047981 -0.0328678 -0.0793534  2.3483450 -0.6888512 -0.1623877  1.5330390
  26250043750 -0.0410180 -0.0558605 -0.1222775 -0.1247990  2.3538200  0.7648772  0.2346988 -1.4048730
  29166707500  0.0288526  0.0439963 -0.0955389 -0.0263796  0.9856992  2.0585430 -0.2075353  1.0147140
  32083371250  0.0489659 -0.0333136 -0.1567631  0.0501767 -0.5467909  2.5521760  0.3111275 -0.5038378
  35000035000 -0.0529191 -0.0062693 -0.0728885  0.1027902 -2.1018580  1.6895910 -0.0470274 -0.2981209
  37916698750 -0.0115174  0.0593715 -0.0154580  0.1516233 -2.8612110  0.3001677  0.1005001  0.2191426
  40833362500  0.0568175 -0.0823666  0.0693232  0.1265148 -2.2649180 -1.3789750  0.3754029 -0.0069353
  43750026250 -0.0709438 -0.0195181  0.1385550  0.0207631 -0.4980423 -2.4883720 -0.9161716 -0.1556719
  46666690000  0.0541739  0.0772672  0.1448704 -0.0082430  1.4939730 -2.4399950  0.9356565  0.2189161
  49583353750 -0.0148967 -0.0357200  0.0580129 -0.0752051  2.7372650 -1.5127880 -0.9971392  0.1729032
  52500017500 -0.0197697 -0.0184779  0.0722836 -0.1085619  2.9212420  0.3330189  0.6627468 -0.5370460
  55416681250  0.0219164  0.0465218 -0.0275210 -0.0922016  1.5701640  2.0137420 -0.6271109 -0.2483641
  58333345000  0.0203718 -0.0610937 -0.0445756 -0.0801138 -0.3243632  3.0771890  0.9611393  0.1363152
  61250008750 -0.0498860  0.0459293 -0.0603399 -0.0396275 -1.7308440 -1.2109820  0.6648751 -0.4649398
  64166672500  0.0445245 -0.0161728 -0.1025364 -0.0495447 -0.5880045 -1.8654230 -0.6092182  0.0381007
  67083336250 -0.0195271 -0.0151763 -0.1001718  0.0447220  0.7933314 -1.5346060  0.5419462  0.4182261
  70000000000  0.0055290  0.0290673 -0.0718908  0.0890640  1.6740700 -0.5463024 -0.0668940 -0.4674752
```

Figure D-88. Transfer of an S2P file to the PC

Example 7 – Output BMP File

A similar technique can be used to get the bitmap data to a file as in the previous example. The Lightning commands “BMPC;OBMP” are used to output a bitmap file. BMPC selects color on white as the color scheme, making for better printouts. The au37xxx_readAsciiARBBlock() function is used to strip off the arbitrary block header and place the bitmap data into a file (the header corrupts the bitmap file).

```
// Include files
#include <ansi_c.h>
#include <visa.h>
#include <userint.h>
#include <formatio.h>
#include "au37xxx.h"

ViSession session;
ViStatus checkErr (ViStatus status);
#define CHECKERR(fCal) \
    if (au37xxx_status = checkErr((fCal)), au37xxx_status < VI_SUCCESS) \
        goto Error; else

int main (int argc, char *argv[])
{
    ViInt32 retCount;
    ViStatus status;
    ViUInt32 read_count;
    ViStatus au37xxx_status = VI_SUCCESS;
    int fileHandle;
    static ViChar readBuffer[600000];

    CHECKERR(au37xxx_init ("VectorStar_Test", VI_FALSE, VI_TRUE, &session));
    CHECKERR(au37xxx_write(session,"LANG NATIVE"));
    CHECKERR(au37xxx_write(session,"BMPC;OBMP"));
```

```
    CHECKERR(au37xxx_readAsciiARBBlock(session,600000,readBuffer,&retCount));
    au37xxx_close(session);

    fileHandle = OpenFile (".\\dave.bmp", VAL_WRITE_ONLY, VAL_OPEN_AS_IS,
VAL_ASCII);
    WriteFile (fileHandle, readBuffer, retCount);
    CloseFile (fileHandle);


Error:
    printf("\n\nHit return to exit:");
    getc(stdin);
    return 0;
}


ViStatus checkErr (ViStatus status)
{
    ViChar  error_message [256];
    ViChar  error_buffer [1024];
    ViUInt16 stb;
    ViUInt16 VNA_ERROR = 4;

    if (status >= 0)
        viReadSTB (session, &stb);

    //check if stb & VNA_ERROR is set

    if ((status < VI_SUCCESS) ||  ((stb & VNA_ERROR) > 0))
    {
        au37xxx_error_message (session, status, error_message);
        sprintf (error_buffer, "Primary Error: 0x%08X, %s\n", status,
error_message);
        printf ("%s\n", error_buffer);
        au37xxx_error_query (session, error_message);
        SetWaitCursor (0);
        sprintf (error_buffer, "Instrument Error: %s\n", error_message);
        printf ("%s\n", error_buffer);
        au37xxx_close(session);
        session = 0;
    }
    return status;
}
```

When the VI runs, a dialog allow the file name to be selected. Make sure to save the file with a “.bmp” extension.

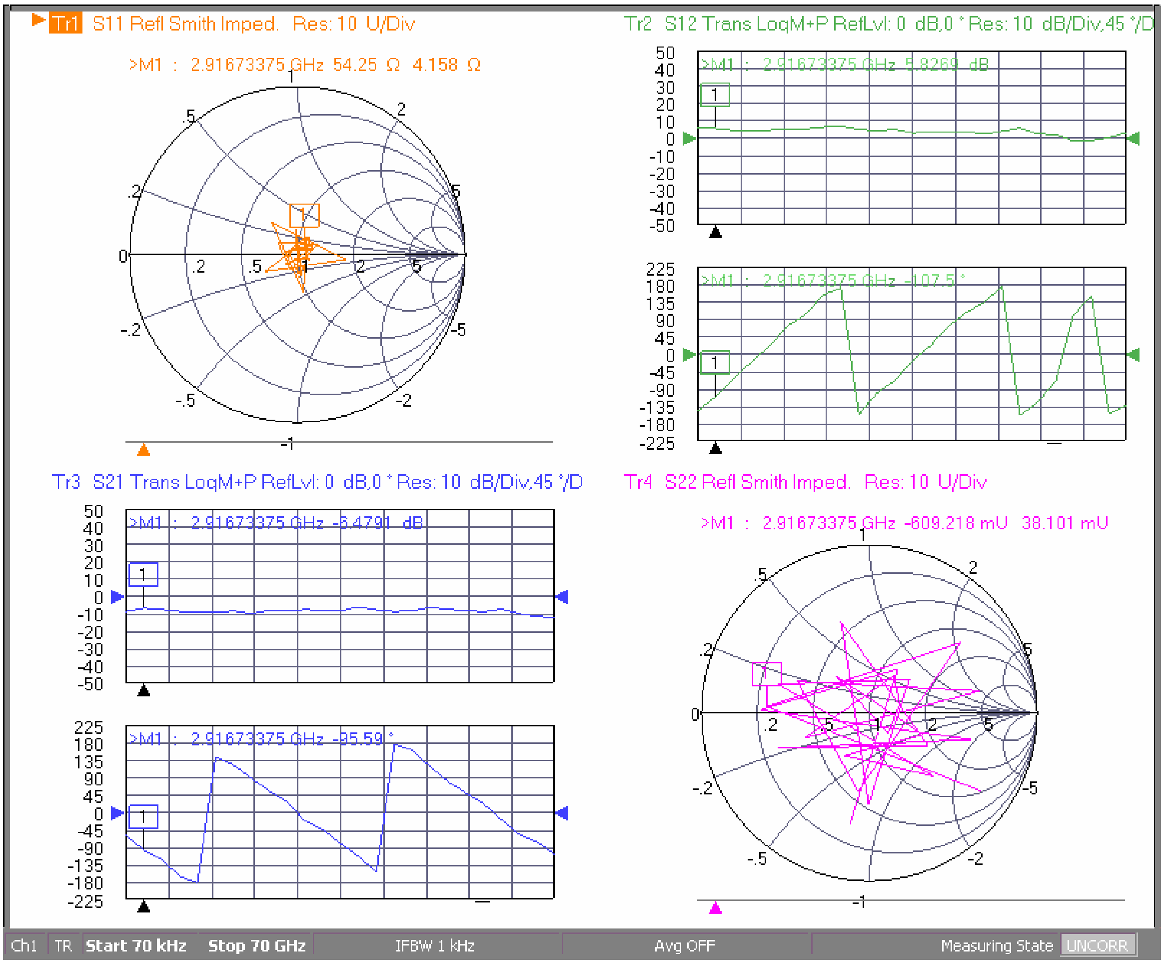


Figure D-89. MS464xB VNA Bitmap Display

Appendix E — Alphabetical Command Index

E-1 Introduction

This appendix provides an alphabetical listing of all commands and queries available for the VectorStar Series VNA.

Sorting

Due to ASCII sorting:

- Commands beginning with a colon (“ : ” or SCPI commands) are sorted first.
- Commands with an optional keyword indicated by brackets (“[ ]”) sort after the non-optional keywords. For example :STATe sorts before [:STATe].
- Commands that start with an asterisk (“ * ” or IEEE 488.2 commands) are sorted next.
- Commands that start with a number or letter are sorted last.

System Suffix

The “System” suffix has been added to the VectorStar Series VNA system, diagnostic, and troubleshooting commands in this index list. These are only for identification and not a part of the actual command syntax.

E-2 Alphabetical Command Listing

:CALCulate:MARKer:TABle[:STATe] <char> 5-76
:CALCulate:MARKer:TABle[:STATe]? 5-76
:CALCulate{1-16}:AFCW:CDURation <NRf> 5-6
:CALCulate{1-16}:AFCW:CDURation? 5-6
:CALCulate{1-16}:AFCW:DELay{1-8} <NRf> 5-7
:CALCulate{1-16}:AFCW:DELay{1-8}? 5-7
:CALCulate{1-16}:AFCW:ESM <char> 5-8
:CALCulate{1-16}:AFCW:ESM? 5-8
:CALCulate{1-16}:AFCW:ISAM <NRf> 5-8
:CALCulate{1-16}:AFCW:ISAM? 5-8
:CALCulate{1-16}:AFCW:ISYN <NRf> 5-9
:CALCulate{1-16}:AFCW:ISYN? 5-9
:CALCulate{1-16}:AFCW:MODe <char> 5-9
:CALCulate{1-16}:AFCW:MODe? 5-9
:CALCulate{1-16}:AFCW:MWIDth{1-8} <NRf> 5-10
:CALCulate{1-16}:AFCW:MWIDth{1-8}? 5-10
:CALCulate{1-16}:AFCW:POINts <NRf> 5-10
:CALCulate{1-16}:AFCW:POINts? 5-10
:CALCulate{1-16}:AFCW:RDEX:BDIRectory <string> 5-11
:CALCulate{1-16}:AFCW:RDEX:BDIRectory? 5-11
:CALCulate{1-16}:AFCW:RDEX:FORMat <char> 5-11
:CALCulate{1-16}:AFCW:RDEX:FORMat? 5-11
:CALCulate{1-16}:AFCW:RECeiver:COUPle[:STATe] <char> 5-11
:CALCulate{1-16}:AFCW:RECeiver:COUPle[:STATe]? 5-11
:CALCulate{1-16}:AFCW:RECeiver:TYPe <char> 5-12
:CALCulate{1-16}:AFCW:RECeiver:TYPe? 5-12
:CALCulate{1-16}:AFCW:STOP:DELay{1-8} <NRf> 5-12
:CALCulate{1-16}:AFCW:STOP:DELay{1-8}? 5-12
:CALCulate{1-16}:APPLication:MEASurement:TYPe <char> 5-13
:CALCulate{1-16}:APPLication:MEASurement:TYPe? 5-13
:CALCulate{1-16}:CORRection:ADAPter:REMoval 5-14
:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:X <string> 5-14
:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:X? 5-14
:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:Y <string> 5-15
:CALCulate{1-16}:CORRection:ADAPter:REMoval:CALibration:Y? 5-15
:CALCulate{1-16}:CORRection:ADAPter:REMoval:LENGth <NRf> 5-15

:CALCulate{1-16}:CORRection:ADAPter:REMoval:LENGth? 5-15
:CALCulate{1-16}:CORRection:MERGe 5-15
:CALCulate{1-16}:CORRection:MERGe:CALibration{1-2} <string> 5-16
:CALCulate{1-16}:CORRection:MERGe:CALibration{1-2}? 5-16
:CALCulate{1-16}:DISPlay:MARKer:ALL[:STATe] <char> 5-17
:CALCulate{1-16}:DISPlay:MARKer:ALL[:STATe]? 5-17
:CALCulate{1-16}:DISPlay:MARKer:INOVerlay[:STATe] <char> 5-17
:CALCulate{1-16}:DISPlay:MARKer:INOVerlay[:STATe]? 5-17
:CALCulate{1-16}:EOOE:EO4Measurment:CALCulate? 5-18
:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile <char> 5-18
:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile:SWAP[:STATe] <char> 5-19
:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile:SWAP[:STATe]? 5-19
:CALCulate{1-16}:EOOE:EO4Measurment:CHARfile? 5-18
:CALCulate{1-16}:EOOE:EO4Measurment:CONFiguration <char> 5-19
:CALCulate{1-16}:EOOE:EO4Measurment:CONFiguration? 5-19
:CALCulate{1-16}:EOOE:EO4Measurment:EOPort <char> 5-19
:CALCulate{1-16}:EOOE:EO4Measurment:EOPort? 5-19
:CALCulate{1-16}:EOOE:EO4Measurment:OEPort <char> 5-20
:CALCulate{1-16}:EOOE:EO4Measurment:OEPort? 5-20
:CALCulate{1-16}:EOOE:EOMeasurment:CALCulate? 5-20
:CALCulate{1-16}:EOOE:EOMeasurment:CALFile <string> 5-21
:CALCulate{1-16}:EOOE:EOMeasurment:CALFile? 5-21
:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile <string> 5-21
:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile:SWAP[:STATe] <char> 5-21
:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile:SWAP[:STATe]? 5-21
:CALCulate{1-16}:EOOE:EOMeasurment:CHARfile? 5-21
:CALCulate{1-16}:EOOE:EOMeasurment:EOPort <char> 5-22
:CALCulate{1-16}:EOOE:EOMeasurment:EOPort? 5-22
:CALCulate{1-16}:EOOE:GO4Measurment:CALCulate? 5-22
:CALCulate{1-16}:EOOE:GO4Measurment:CALFile <string> 5-23
:CALCulate{1-16}:EOOE:GO4Measurment:CALFile? 5-23
:CALCulate{1-16}:EOOE:GO4Measurment:CHARfile <char> 5-23
:CALCulate{1-16}:EOOE:GO4Measurment:CHARfile? 5-23
:CALCulate{1-16}:EOOE:GO4Measurment:CONFiguration <char> 5-23
:CALCulate{1-16}:EOOE:GO4Measurment:CONFiguration? 5-23
:CALCulate{1-16}:EOOE:GO4Measurment:EOPort <char> 5-24
:CALCulate{1-16}:EOOE:GO4Measurment:EOPort? 5-24
:CALCulate{1-16}:EOOE:GO4Measurment:OEPort <char> 5-24
:CALCulate{1-16}:EOOE:GO4Measurment:OEPort? 5-24
:CALCulate{1-16}:EOOE:GO4Measurment:TARGetfile <string> 5-24
:CALCulate{1-16}:EOOE:GO4Measurment:TARGetfile? 5-24
:CALCulate{1-16}:EOOE:GOMeasurment:CALCulate? 5-25
:CALCulate{1-16}:EOOE:GOMeasurment:CALFile <string> 5-25
:CALCulate{1-16}:EOOE:GOMeasurment:CALFile? 5-25
:CALCulate{1-16}:EOOE:GOMeasurment:CHARfile <char> 5-25
:CALCulate{1-16}:EOOE:GOMeasurment:CHARfile? 5-25
:CALCulate{1-16}:EOOE:GOMeasurment:EOPort <char> 5-25
:CALCulate{1-16}:EOOE:GOMeasurment:EOPort? 5-25
:CALCulate{1-16}:EOOE:GOMeasurment:TARGetfile <string> 5-26
:CALCulate{1-16}:EOOE:GOMeasurment:TARGetfile? 5-26
:CALCulate{1-16}:EOOE:MSGS:LIST? 5-26
:CALCulate{1-16}:EOOE:OE4Measurment:CALCulate? 5-27
:CALCulate{1-16}:EOOE:OE4Measurment:CALFile <string> 5-27
:CALCulate{1-16}:EOOE:OE4Measurment:CALFile? 5-27
:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile <string> 5-28
:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile:SWAP[:STATe] <char> 5-28
:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile:SWAP[:STATe]? 5-28
:CALCulate{1-16}:EOOE:OE4Measurment:CHARfile? 5-28
:CALCulate{1-16}:EOOE:OE4Measurment:CONFiguration <char> 5-28
:CALCulate{1-16}:EOOE:OE4Measurment:CONFiguration? 5-28
:CALCulate{1-16}:EOOE:OE4Measurment:EOPort <char> 5-29
:CALCulate{1-16}:EOOE:OE4Measurment:EOPort? 5-29
:CALCulate{1-16}:EOOE:OE4Measurment:OEPort <char> 5-29
:CALCulate{1-16}:EOOE:OE4Measurment:OEPort? 5-29
:CALCulate{1-16}:EOOE:OEMeasurment:CALCulate? 5-30
:CALCulate{1-16}:EOOE:OEMeasurment:CALFile <string> 5-30
:CALCulate{1-16}:EOOE:OEMeasurment:CALFile? 5-30

:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile <string> 5-31
:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile:SWAP[:STATe] <char> 5-31
:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile:SWAP[:STATe]? 5-31
:CALCulate{1-16}:EOOE:OEMeasurment:CHARfile? 5-31
:CALCulate{1-16}:EOOE:OEMeasurment:EOPort <char> 5-31
:CALCulate{1-16}:EOOE:OEMeasurment:EOPort? 5-31
:CALCulate{1-16}:EOOE:TYPE? 5-32
:CALCulate{1-16}:EXTRaction 5-33
:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe <string> 5-34
:CALCulate{1-16}:EXTRaction:CALibration:CALB:FILe? 5-34
:CALCulate{1-16}:EXTRaction:CALibration:CALB:PORT <char> 5-35
:CALCulate{1-16}:EXTRaction:CALibration:CALB:PORT? 5-35
:CALCulate{1-16}:EXTRaction:CALibration:INNer <string> 5-35
:CALCulate{1-16}:EXTRaction:CALibration:INNer? 5-35
:CALCulate{1-16}:EXTRaction:CALibration:OUTer <string> 5-35
:CALCulate{1-16}:EXTRaction:CALibration:OUTer? 5-35
:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:FILe <string> 5-33
:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:FILe? 5-33
:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:PORT <char> 5-34
:CALCulate{1-16}:EXTRaction:CALibration[:CALa]:PORT? 5-34
:CALCulate{1-16}:EXTRaction:ELL1:LENGth <NRf> 5-36
:CALCulate{1-16}:EXTRaction:ELL1:LENGth? 5-36
:CALCulate{1-16}:EXTRaction:ELL2:LENGth <NRf> 5-36
:CALCulate{1-16}:EXTRaction:ELL2:LENGth? 5-36
:CALCulate{1-16}:EXTRaction:ELL3:LENGth <NRf> 5-37
:CALCulate{1-16}:EXTRaction:ELL3:LENGth? 5-37
:CALCulate{1-16}:EXTRaction:ELL4:LENGth <NRf> 5-37
:CALCulate{1-16}:EXTRaction:ELL4:LENGth? 5-37
:CALCulate{1-16}:EXTRaction:S2P1filename:FILe <string> 5-37
:CALCulate{1-16}:EXTRaction:S2P1filename:FILe? 5-37
:CALCulate{1-16}:EXTRaction:S2P2filename:FILe <string> 5-38
:CALCulate{1-16}:EXTRaction:S2P2filename:FILe? 5-38
:CALCulate{1-16}:EXTRaction:S2P3filename:FILe <string> 5-38
:CALCulate{1-16}:EXTRaction:S2P3filename:FILe? 5-38
:CALCulate{1-16}:EXTRaction:S2P4filename:FILe <string> 5-38
:CALCulate{1-16}:EXTRaction:S2P4filename:FILe? 5-38
:CALCulate{1-16}:EXTRaction:S4P1filename:FILe <string> 5-39
:CALCulate{1-16}:EXTRaction:S4P1filename:FILe? 5-39
:CALCulate{1-16}:EXTRaction:S4P2filename:FILe <string> 5-39
:CALCulate{1-16}:EXTRaction:S4P2filename:FILe? 5-39
:CALCulate{1-16}:EXTRaction:SXPPortpair:PORT <char> 5-39
:CALCulate{1-16}:EXTRaction:SXPPortpair:PORT? 5-39
:CALCulate{1-16}:EXTRaction:ZERo:MATCh[:STATe] <char> 5-40
:CALCulate{1-16}:EXTRaction:ZERo:MATCh[:STATe]? 5-40
:CALCulate{1-16}:EXTRaction[:METHod]:A 5-40
:CALCulate{1-16}:EXTRaction[:METHod]:B 5-41
:CALCulate{1-16}:EXTRaction[:METHod]:C 5-41
:CALCulate{1-16}:EXTRaction[:METHod]:D 5-42
:CALCulate{1-16}:EXTRaction[:METHod]:E 5-42
:CALCulate{1-16}:EXTRaction[:METHod]:F 5-43
:CALCulate{1-16}:EXTRaction[:METHod]:G 5-43
:CALCulate{1-16}:FCW:CPCount? 5-44
:CALCulate{1-16}:FCW:DATA? 5-44
:CALCulate{1-16}:FCW:DCOLlect <char> 5-45
:CALCulate{1-16}:FCW:DCOLlect? 5-45
:CALCulate{1-16}:FCW:DTYPe <char> 5-45
:CALCulate{1-16}:FCW:DTYPe? 5-45
:CALCulate{1-16}:FCW:IBUF:POINts <NRf> 5-46
:CALCulate{1-16}:FCW:IBUF:POINts? 5-46
:CALCulate{1-16}:FCW:MARK <char> 5-46
:CALCulate{1-16}:FCW:MODE <char> 5-47
:CALCulate{1-16}:FCW:MODE? 5-47
:CALCulate{1-16}:FCW:MPCount? 5-47
:CALCulate{1-16}:FCW:STReam:POINts <NRf> 5-48
:CALCulate{1-16}:FCW:STReam:POINts? 5-48
:CALCulate{1-16}:FCW[:STATe] <char> 5-47
:CALCulate{1-16}:FCW[:STATe]? 5-47

:CALCulate{1-16}:FORMat:S1P:PORT <char> . . . 5-49
:CALCulate{1-16}:FORMat:S1P:PORT? . . . 5-49
:CALCulate{1-16}:FORMat:S2P:PORT <char> . . . 5-49
:CALCulate{1-16}:FORMat:S2P:PORT? . . . 5-49
:CALCulate{1-16}:FORMat:S3P:PORT <char> . . . 5-50
:CALCulate{1-16}:FORMat:S3P:PORT? . . . 5-50
:CALCulate{1-16}:FSIMulator:N . . . 5-66
:CALCulate{1-16}:FSIMulator:NETWork:ADD . . . 5-51
:CALCulate{1-16}:FSIMulator:NETWork:C <NRf> . . . 5-51
:CALCulate{1-16}:FSIMulator:NETWork:C? . . . 5-51
:CALCulate{1-16}:FSIMulator:NETWork:CLEar . . . 5-51
:CALCulate{1-16}:FSIMulator:NETWork:COUNt? . . . 5-52
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric <NRf> . . . 5-52
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:EVEN <NRf> . . . 5-53
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:EVEN? . . . 5-53
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:ODD <NRf> . . . 5-53
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric:ODD? . . . 5-53
:CALCulate{1-16}:FSIMulator:NETWork:DIELectric? . . . 5-52
:CALCulate{1-16}:FSIMulator:NETWork:FREQuency <NRf> . . . 5-54
:CALCulate{1-16}:FSIMulator:NETWork:FREQuency? . . . 5-54
:CALCulate{1-16}:FSIMulator:NETWork:L <NRf> . . . 5-54
:CALCulate{1-16}:FSIMulator:NETWork:L? . . . 5-54
:CALCulate{1-16}:FSIMulator:NETWork:LENGth <NRf> . . . 5-54
:CALCulate{1-16}:FSIMulator:NETWork:LENGth? . . . 5-54
:CALCulate{1-16}:FSIMulator:NETWork:LOSS <NRf> . . . 5-55
:CALCulate{1-16}:FSIMulator:NETWork:LOSS:EVEN <NRf> . . . 5-55
:CALCulate{1-16}:FSIMulator:NETWork:LOSS:EVEN? . . . 5-55
:CALCulate{1-16}:FSIMulator:NETWork:LOSS:ODD <NRf> . . . 5-56
:CALCulate{1-16}:FSIMulator:NETWork:LOSS:ODD? . . . 5-56
:CALCulate{1-16}:FSIMulator:NETWork:LOSS? . . . 5-55
:CALCulate{1-16}:FSIMulator:NETWork:MODe <char> . . . 5-56
:CALCulate{1-16}:FSIMulator:NETWork:MODe? . . . 5-56
:CALCulate{1-16}:FSIMulator:NETWork:PORT <char> . . . 5-57
:CALCulate{1-16}:FSIMulator:NETWork:PORT? . . . 5-57
:CALCulate{1-16}:FSIMulator:NETWork:R <NRf> . . . 5-57
:CALCulate{1-16}:FSIMulator:NETWork:R? . . . 5-57
:CALCulate{1-16}:FSIMulator:NETWork:S2P <string> . . . 5-57
:CALCulate{1-16}:FSIMulator:NETWork:S2P? . . . 5-57
:CALCulate{1-16}:FSIMulator:NETWork:S4P <string> . . . 5-58
:CALCulate{1-16}:FSIMulator:NETWork:S4P:PORTs <char1>, <char2>, <char3>, <char4> . . . 5-58
:CALCulate{1-16}:FSIMulator:NETWork:S4P:PORTs? . . . 5-58
:CALCulate{1-16}:FSIMulator:NETWork:S4P:TERM:IGNore <char1>, {<char2>, ..., <charn>} . . . 5-59
:CALCulate{1-16}:FSIMulator:NETWork:S4P:TERM:IGNore? . . . 5-59
:CALCulate{1-16}:FSIMulator:NETWork:S4P:TRANsmission:TERM <NRf> . . . 5-59
:CALCulate{1-16}:FSIMulator:NETWork:S4P:TRANsmission:TERM? . . . 5-59
:CALCulate{1-16}:FSIMulator:NETWork:S4P? . . . 5-58
:CALCulate{1-16}:FSIMulator:NETWork:SWAPs2p <char> . . . 5-59
:CALCulate{1-16}:FSIMulator:NETWork:SWAPs2p? . . . 5-59
:CALCulate{1-16}:FSIMulator:NETWork:TYPe <char> . . . 5-60
:CALCulate{1-16}:FSIMulator:NETWork:TYPe? . . . 5-60
:CALCulate{1-16}:FSIMulator:NETWork:Z0 <NRf> . . . 5-61
:CALCulate{1-16}:FSIMulator:NETWork:Z0:EVEN <NRf> . . . 5-61
:CALCulate{1-16}:FSIMulator:NETWork:Z0:EVEN? . . . 5-61
:CALCulate{1-16}:FSIMulator:NETWork:Z0:ODD <NRf> . . . 5-62
:CALCulate{1-16}:FSIMulator:NETWork:Z0:ODD? . . . 5-62
:CALCulate{1-16}:FSIMulator:NETWork:Z0? . . . 5-61
:CALCulate{1-16}:FSIMulator:NETWork[:STATe] <char> . . . 5-62
:CALCulate{1-16}:FSIMulator:NETWork[:STATe]? . . . 5-62
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:C <NRf> . . . 5-63
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:C? . . . 5-63
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DELete . . . 5-63
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric <NRf> . . . 5-64
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:EVEN <NRf> . . . 5-64
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:EVEN? . . . 5-64
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:ODD <NRf> . . . 5-65
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric:ODD? . . . 5-65
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:DIELectric? . . . 5-64

:CALCulate{1-16}:FSIMulator:NETWork{1-50}:FREQuency <NRf> . . . 5-65
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:FREQuency? . . . 5-65
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:L <NRf> . . . 5-66
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LENGth <NRf> . . . 5-66
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LENGth? . . . 5-66
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS <NRf> . . . 5-66
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:EVEN <NRf> . . . 5-67
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:EVEN? . . . 5-67
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:ODD <NRf> . . . 5-67
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS:ODD? . . . 5-67
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:LOSS? . . . 5-66
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:MODe <char> . . . 5-68
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:MODe? . . . 5-68
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:PORT <char> . . . 5-68
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:PORT? . . . 5-68
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:R <NRf> . . . 5-68
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:R? . . . 5-68
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S2P <string> . . . 5-69
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S2P? . . . 5-69
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P <string> . . . 5-69
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:PORTs <char>, <char>, <char>, <char> . . . 5-69
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:PORTs? . . . 5-69
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TERM:IGNore <char> {,<char2>, ...,<charn>} . . . 5-70
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TERM:IGNore? . . . 5-70
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TRANsmission:TERM <NRf> . . . 5-70
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P:TRANsmission:TERM? . . . 5-70
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:S4P? . . . 5-69
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:SWAPs2p <char> . . . 5-70
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:SWAPs2p? . . . 5-70
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:TYPe <char> . . . 5-71
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:TYPe? . . . 5-71
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0 <NRf> . . . 5-72
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:EVEN <NRf> . . . 5-72
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:EVEN? . . . 5-72
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:ODD <NRf> . . . 5-73
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0:ODD? . . . 5-73
:CALCulate{1-16}:FSIMulator:NETWork{1-50}:Z0? . . . 5-72
:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:R0 <NRf> . . . 5-74
:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:R0? . . . 5-74
:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:X0 <NRf> . . . 5-74
:CALCulate{1-16}:IMPedance:TRANsformation:PORT{1-4}:X0? . . . 5-74
:CALCulate{1-16}:IMPedance:TRANsformation[:STATe] . . . 5-75
:CALCulate{1-16}:IMPedance:TRANsformation[:STATe]? . . . 5-75
:CALCulate{1-16}:MARKer:COUPle <char> . . . 5-76
:CALCulate{1-16}:MARKer:COUPle? . . . 5-76
:CALCulate{1-16}:MXP:D1S0:TOPology <char> . . . 5-77
:CALCulate{1-16}:MXP:D1S0:TOPology? . . . 5-77
:CALCulate{1-16}:MXP:D1S1:TOPology <char1>, <char2> . . . 5-78
:CALCulate{1-16}:MXP:D1S1:TOPology? . . . 5-78
:CALCulate{1-16}:MXP:D1S2:TOPology <char1>, <char2>, <char3> . . . 5-79
:CALCulate{1-16}:MXP:D1S2:TOPology? . . . 5-79
:CALCulate{1-16}:MXP:D2S0:TOPology <char1>, <char2> . . . 5-81
:CALCulate{1-16}:MXP:D2S0:TOPology? . . . 5-81
:CALCulate{1-16}:NFIGure:LOAD:DUT:S2P:DATA <char> . . . 5-83
:CALCulate{1-16}:NFIGure:LOAD:RECVer:OFFSet:DATA <filespec> . . . 5-84
:CALCulate{1-16}:NFIGure:NETWork:DEEMbed:FILe <string> . . . 5-85
:CALCulate{1-16}:NFIGure:NETWork:DEEMbed:FILe? . . . 5-85
:CALCulate{1-16}:NFIGure:NETWork:DEEMbed:VALid? . . . 5-85
:CALCulate{1-16}:NFIGure:NETWork:DEEMbed[:STATe] <char> . . . 5-85
:CALCulate{1-16}:NFIGure:NETWork:DEEMbed[:STATe]? . . . 5-85
:CALCulate{1-16}:NFIGure:NETWork:EMBed:FILe <string> . . . 5-86
:CALCulate{1-16}:NFIGure:NETWork:EMBed:FILe? . . . 5-86
:CALCulate{1-16}:NFIGure:NETWork:EMBed:VALid? . . . 5-86
:CALCulate{1-16}:NFIGure:NETWork:EMBed[:STATe] <char> . . . 5-86
:CALCulate{1-16}:NFIGure:NETWork:EMBed[:STATe]? . . . 5-86
:CALCulate{1-16}:NFIGure:NOISe:CALibration:PERForm? . . . 5-83
:CALCulate{1-16}:NFIGure:RECVer:OFFSet[:STATe] <char> . . . 5-87

:CALCulate{1-16}:NFIGure:RECVer:OFFSet[:STATe]? . . . 5-87
:CALCulate{1-16}:NFIGure:RMS:POINts <NRf> . . . 5-84
:CALCulate{1-16}:NFIGure:RMS:POINts? . . . 5-84
:CALCulate{1-16}:NFIGure:TEMPerature? . . . 5-84
:CALCulate{1-16}:NXN:FILe <char>, <string> . . . 5-88
:CALCulate{1-16}:NXN:FILe? . . . 5-88
:CALCulate{1-16}:NXN:IFPath:DEEMbedding[:STATe] <char> . . . 5-89
:CALCulate{1-16}:NXN:IFPath:DEEMbedding[:STATe]? . . . 5-89
:CALCulate{1-16}:NXN:IFPath:DIRection <char> . . . 5-89
:CALCulate{1-16}:NXN:IFPath:DIRection? . . . 5-89
:CALCulate{1-16}:NXN:IFPath:FILe <string> . . . 5-89
:CALCulate{1-16}:NXN:IFPath:FILe? . . . 5-89
:CALCulate{1-16}:NXN:IFPath:MODified:FILe <string> . . . 5-90
:CALCulate{1-16}:NXN:IFPath:MODified:FILe? . . . 5-90
:CALCulate{1-16}:NXN:IFPath:MODified:SAVe . . . 5-90
:CALCulate{1-16}:NXN:LENGth <char>, <NRf> . . . 5-90
:CALCulate{1-16}:NXN:LENGth? . . . 5-90
:CALCulate{1-16}:NXN:SOLVe <char> . . . 5-91
:CALCulate{1-16}:PARameter:COUNt <NRf> . . . 5-92
:CALCulate{1-16}:PARameter:COUNt? . . . 5-92
:CALCulate{1-16}:PARameter:SELect? . . . 5-98
:CALCulate{1-16}:PARameter{1-16}:DEFine <char1> | <char1>,<char2> | <char1>,<char2>,<char3>,<char4> . . . 5-93
:CALCulate{1-16}:PARameter{1-16}:DEFine? . . . 5-93
:CALCulate{1-16}:PARameter{1-16}:FORMat <char> . . . 5-97
:CALCulate{1-16}:PARameter{1-16}:FORMat? . . . 5-97
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:DEFine <char> . . . 5-101
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:DEFine? . . . 5-101
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:TOPology <char> . . . 5-101
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S0:TOPology? . . . 5-101
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:DEFine <char> . . . 5-102
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:DEFine? . . . 5-102
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:TOPology <char1>, <char2> . . . 5-102
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S1:TOPology? . . . 5-102
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:DEFine <char> . . . 5-103
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:DEFine? . . . 5-103
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:TOPology <char1>, <char2>, <char3> . . . 5-103
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D1S2:TOPology? . . . 5-103
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:DEFine <char> . . . 5-105
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:DEFine? . . . 5-105
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:TOPology <char1>, <char2> . . . 5-106
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:D2S0:TOPology? . . . 5-106
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:DEVice <char> . . . 5-106
:CALCulate{1-16}:PARameter{1-16}:FSIMulator:BALun:DEVice? . . . 5-106
:CALCulate{1-16}:PARameter{1-16}:MARKer:ACTivate? . . . 5-107
:CALCulate{1-16}:PARameter{1-16}:MARKer:DISCrete <char> . . . 5-107
:CALCulate{1-16}:PARameter{1-16}:MARKer:DISCrete? . . . 5-107
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:ACTivate . . . 5-108
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:X <NRf> . . . 5-108
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:X? . . . 5-108
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}:Y? . . . 5-108
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}[:STATe] <char> . . . 5-109
:CALCulate{1-16}:PARameter{1-16}:MARKer{1-13}[:STATe]? . . . 5-109
:CALCulate{1-16}:PARameter{1-16}:MLOCation <char> . . . 5-98
:CALCulate{1-16}:PARameter{1-16}:MLOCation? . . . 5-98
:CALCulate{1-16}:PARameter{1-16}:MSTatistics <char> . . . 5-110
:CALCulate{1-16}:PARameter{1-16}:MSTatistics:DATA? . . . 5-110
:CALCulate{1-16}:PARameter{1-16}:MSTatistics:DATA2? . . . 5-111
:CALCulate{1-16}:PARameter{1-16}:MSTatistics? . . . 5-110
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:AUTOmatic . . . 5-99
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:DISTance <NRf> . . . 5-99
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:DISTance? . . . 5-99
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:LOSS <NRf> . . . 5-99
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:LOSS? . . . 5-99
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:PHAse <NRf> . . . 5-100
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:PHAse? . . . 5-100
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:TIMe <NRf> . . . 5-100
:CALCulate{1-16}:PARameter{1-16}:REFerence:EXTension:TIMe? . . . 5-100

:CALCulate{1-16}:PARameter{1-16}:SELect . . . 5-112
:CALCulate{1-16}:POLar:ANGLe:STARt <NRf> . . . 5-113
:CALCulate{1-16}:POLar:ANGLe:STARt? . . . 5-113
:CALCulate{1-16}:POLar:ANGLe:STOP <NRf> . . . 5-114
:CALCulate{1-16}:POLar:ANGLe:STOP? . . . 5-114
:CALCulate{1-16}:POLar:CHARt <char> . . . 5-114
:CALCulate{1-16}:POLar:CHARt? . . . 5-114
:CALCulate{1-16}:PROCessing:ORDer:GRPDelay <char> . . . 5-115
:CALCulate{1-16}:PROCessing:ORDer:GRPDelay? . . . 5-115
:CALCulate{1-16}:PROCessing:ORDer:REFPlane <char> . . . 5-115
:CALCulate{1-16}:PROCessing:ORDer:REFPlane? . . . 5-115
:CALCulate{1-16}:PULSe:CONFig:DEFine:DONe:INFo? . . . 5-116
:CALCulate{1-16}:PULSe:CONFig:DEFine[:STATe] <char> . . . 5-116
:CALCulate{1-16}:PULSe:CONFig:DEFine[:STATe]? . . . 5-116
:CALCulate{1-16}:PULSe:CONTinuous:CACType <char> . . . 5-117
:CALCulate{1-16}:PULSe:CONTinuous:CACType? . . . 5-117
:CALCulate{1-16}:PULSe:CONTinuous:CCDuration? . . . 5-117
:CALCulate{1-16}:PULSe:CONTinuous:DCDuration <NRf> . . . 5-117
:CALCulate{1-16}:PULSe:CONTinuous:DCDuration? . . . 5-117
:CALCulate{1-16}:PULSe:CONTinuous:PPOints <NRf> . . . 5-118
:CALCulate{1-16}:PULSe:CONTinuous:PPOints? . . . 5-118
:CALCulate{1-16}:PULSe:CONTinuous:UDCDuration[:STATe] <char> . . . 5-118
:CALCulate{1-16}:PULSe:CONTinuous:UDCDuration[:STATe]? . . . 5-118
:CALCulate{1-16}:PULSe:COUPle:RECeiver:PARameter[:STATe] <char> . . . 5-118
:CALCulate{1-16}:PULSe:COUPle:RECeiver:PARameter[:STATe]? . . . 5-118
:CALCulate{1-16}:PULSe:EXTernal:SYNC:MARKing[:STATe] . . . 5-119
:CALCulate{1-16}:PULSe:EXTernal:SYNC:MARKing[:STATe]? . . . 5-119
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:DELay <NRf> . . . 5-119
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:DELay? . . . 5-119
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:NUMBer <NRf> . . . 5-119
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:NUMBer? . . . 5-119
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:PERiod <NRf> . . . 5-119
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:PERiod? . . . 5-119
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:WIDTh <NRf> . . . 5-120
:CALCulate{1-16}:PULSe:GENerator{1-4}:BURSt:WIDTh? . . . 5-120
:CALCulate{1-16}:PULSe:GENerator{1-4}:DUTy:CYCLe? . . . 5-120
:CALCulate{1-16}:PULSe:GENerator{1-4}:ENABle <char> . . . 5-120
:CALCulate{1-16}:PULSe:GENerator{1-4}:ENABle? . . . 5-120
:CALCulate{1-16}:PULSe:GENerator{1-4}:INVerted:POLarity <char> . . . 5-120
:CALCulate{1-16}:PULSe:GENerator{1-4}:INVerted:POLarity? . . . 5-120
:CALCulate{1-16}:PULSe:GENerator{1-4}:LABel <string> . . . 5-121
:CALCulate{1-16}:PULSe:GENerator{1-4}:LABel? . . . 5-121
:CALCulate{1-16}:PULSe:GENerator{1-4}:MODe <char> . . . 5-121
:CALCulate{1-16}:PULSe:GENerator{1-4}:MODe? . . . 5-121
:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:DELay <NRf> . . . 5-121
:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:DELay? . . . 5-121
:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:WIDTh <NRf> . . . 5-122
:CALCulate{1-16}:PULSe:GENerator{1-4}:PULSe{1-4}:WIDTh? . . . 5-122
:CALCulate{1-16}:PULSe:MEASure:GENenerator . . . 5-122
:CALCulate{1-16}:PULSe:MEASure:GENenerator? . . . 5-122
:CALCulate{1-16}:PULSe:MEASure:WIDTh <NRf> . . . 5-122
:CALCulate{1-16}:PULSe:MEASure:WIDTh? . . . 5-122
:CALCulate{1-16}:PULSe:MODe <char> . . . 5-123
:CALCulate{1-16}:PULSe:MODe? . . . 5-123
:CALCulate{1-16}:PULSe:POINts <NRf> . . . 5-123
:CALCulate{1-16}:PULSe:POINts? . . . 5-123
:CALCulate{1-16}:PULSe:PRF <NRf> . . . 5-124
:CALCulate{1-16}:PULSe:PRF? . . . 5-124
:CALCulate{1-16}:PULSe:PRI <NRf> . . . 5-124
:CALCulate{1-16}:PULSe:PRI? . . . 5-124
:CALCulate{1-16}:PULSe:PULSes <NRf> . . . 5-124
:CALCulate{1-16}:PULSe:PULSes? . . . 5-124
:CALCulate{1-16}:PULSe:RECeiver:TYPe <char> . . . 5-125
:CALCulate{1-16}:PULSe:RECeiver:TYPe? . . . 5-125
:CALCulate{1-16}:PULSe:RESolution <NRf> . . . 5-125
:CALCulate{1-16}:PULSe:RESolution? . . . 5-125
:CALCulate{1-16}:PULSe:STARt:DELay <NRf> . . . 5-125

:CALCulate{1-16}:PULSe:STARt:DELay? . . . 5-125
:CALCulate{1-16}:PULSe:STOP:DELay <NRf> . . . 5-126
:CALCulate{1-16}:PULSe:STOP:DELay? . . . 5-126
:CALCulate{1-16}:PULSe:SYNC:TYPe <char> . . . 5-126
:CALCulate{1-16}:PULSe:SYNC:TYPe? . . . 5-126
:CALCulate{1-16}:PULSe:ZOOM:MARKer[:STATe] <char> . . . 5-126
:CALCulate{1-16}:PULSe:ZOOM:MARKer[:STATe]? . . . 5-126
:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric <char> . . . 5-127
:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric:OTHer <NRf> . . . 5-128
:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric:OTHer? . . . 5-128
:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric:VALue? . . . 5-128
:CALCulate{1-16}:REFerence:EXTension:COAXial:DIELectric? . . . 5-127
:CALCulate{1-16}:REFerence:EXTension:LINE <char> . . . 5-128
:CALCulate{1-16}:REFerence:EXTension:LINE? . . . 5-128
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:DIELectric <NRf> . . . 5-129
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:DIELectric? . . . 5-129
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:EFFective <NRf> . . . 5-129
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:EFFective? . . . 5-129
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:THICKness <NRf> . . . 5-129
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:THICKness? . . . 5-129
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:WIDth <NRf> . . . 5-130
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:WIDth? . . . 5-130
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:Z0 <NRf> . . . 5-130
:CALCulate{1-16}:REFerence:EXTension:MICrostrip:Z0? . . . 5-130
:CALCulate{1-16}:REFerence:EXTension:PARameter <char> . . . 5-130
:CALCulate{1-16}:REFerence:EXTension:PARameter? . . . 5-130
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:AUTOmatic . . . 5-131
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:DISTance <NRf> . . . 5-131
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:DISTance? . . . 5-131
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:LOSS <NRf> . . . 5-131
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:LOSS? . . . 5-131
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:PHAse <NRf> . . . 5-132
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:PHAse? . . . 5-132
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:TIMe <NRf> . . . 5-132
:CALCulate{1-16}:REFerence:EXTension:PORT{1-4}:TIMe? . . . 5-132
:CALCulate{1-16}:REFerence:EXTension:WAVeguide:DIELectric <NRf> . . . 5-132
:CALCulate{1-16}:REFerence:EXTension:WAVeguide:DIELectric? . . . 5-132
:CALCulate{1-16}:REFerence:EXTension:WAVeguide:FREQuency <NRf> . . . 5-133
:CALCulate{1-16}:REFerence:EXTension:WAVeguide:FREQuency? . . . 5-133
:CALCulate{1-16}:SNPSetup:GATed:DATa[:STATe] . . . 5-212
:CALCulate{1-16}:SNPSetup:GATed:DATa[:STATe] <char> . . . 5-212
:CALCulate{1-16}:TMS:MODe <char> . . . 5-213
:CALCulate{1-16}:TMS:MODe? . . . 5-213
:CALCulate{1-16}:TMS:PHASe:SWEep:POINts <NRf> . . . 5-213
:CALCulate{1-16}:TMS:PHASe:SWEep:POINts? . . . 5-213
:CALCulate{1-16}:TMS:PHASe:SWEep:STAR? . . . 5-214
:CALCulate{1-16}:TMS:PHASe:SWEep:STARt <NRf> . . . 5-214
:CALCulate{1-16}:TMS:PHASe:SWEep:STOP <NRf> . . . 5-214
:CALCulate{1-16}:TMS:PHASe:SWEep:STOP? . . . 5-214
:CALCulate{1-16}:TMS:PHASe:SWEep[:STATe] <char> . . . 5-213
:CALCulate{1-16}:TMS:PHASe:SWEep[:STATe]? . . . 5-213
:CALCulate{1-16}:TMS:POWer:OFFSet <NRf> . . . 5-214
:CALCulate{1-16}:TMS:POWer:OFFSet? . . . 5-214
:CALCulate{1-16}:TMS:REFerence:PORT <char> . . . 5-214
:CALCulate{1-16}:TMS:REFerence:PORT? . . . 5-214
:CALCulate{1-16}[:SELected]:CONVersion:FUNCtion <char> . . . 5-134
:CALCulate{1-16}[:SELected]:CONVersion:FUNCtion? . . . 5-134
:CALCulate{1-16}[:SELected]:CONVersion[:STATe] <char> . . . 5-135
:CALCulate{1-16}[:SELected]:CONVersion[:STATe]? . . . 5-135
:CALCulate{1-16}[:SELected]:DATA:FDATa <block> . . . 5-136
:CALCulate{1-16}[:SELected]:DATA:FDATa? . . . 5-136
:CALCulate{1-16}[:SELected]:DATA:FMEMory <block> . . . 5-137
:CALCulate{1-16}[:SELected]:DATA:FMEMory? . . . 5-137
:CALCulate{1-16}[:SELected]:DATA:SDATa <Block> . . . 5-137
:CALCulate{1-16}[:SELected]:DATA:SDATa? . . . 5-137
:CALCulate{1-16}[:SELected]:DATA:SMEMory <Block> . . . 5-137
:CALCulate{1-16}[:SELected]:DATA:SMEMory? . . . 5-137

:CALCulate{1-16}[:SELected]:EYE:EXECute 5-138
:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:LENGth <int> 5-139
:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:LENGth? 5-139
:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:TYPE <char> 5-138
:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:TYPE? 5-138
:CALCulate{1-16}[:SELected]:EYE:INPut:BPATtern:USER <string> 5-139
:CALCulate{1-16}[:SELected]:EYE:INPut:DRATe <NRF> 5-140
:CALCulate{1-16}[:SELected]:EYE:INPut:DRATe? 5-140
:CALCulate{1-16}[:SELected]:EYE:INPut:FTIMe:DATA <NRF> 5-141
:CALCulate{1-16}[:SELected]:EYE:INPut:FTIMe:DATA? 5-141
:CALCulate{1-16}[:SELected]:EYE:INPut:HLEVel <NRF> 5-141
:CALCulate{1-16}[:SELected]:EYE:INPut:HLEVel? 5-141
:CALCulate{1-16}[:SELected]:EYE:INPut:HSHift <NRF> 5-140
:CALCulate{1-16}[:SELected]:EYE:INPut:HSHift? 5-140
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:OFFSet <NRF> 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:OFFSet? 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:PROBability <NRF> 5-143
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}:PROBability? 5-143
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}[:STATe] <char> 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:DIRac{1-2}[:STATe]? 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom:MAGNitude <NRF> 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom:MAGNitude? 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom[:STATe] <char> 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:RANDom[:STATe]? 5-142
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:AMPLitude <NRF> 5-144
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:AMPLitude? 5-144
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:FREQuency <NRF> 5-144
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}:FREQuency? 5-144
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}[:STATe] <char> 5-143
:CALCulate{1-16}[:SELected]:EYE:INPut:JITTer:SINusoidal{1-2}[:STATe]? 5-143
:CALCulate{1-16}[:SELected]:EYE:INPut:LLEVel <NRF> 5-141
:CALCulate{1-16}[:SELected]:EYE:INPut:LLEVel? 5-141
:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe:AMPLitude <NRF> 5-145
:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe:AMPLitude? 5-145
:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe[:STATe] <char> 5-145
:CALCulate{1-16}[:SELected]:EYE:INPut:NOISe[:STATe]? 5-145
:CALCulate{1-16}[:SELected]:EYE:INPut:PERSistence <NR1> 5-145
:CALCulate{1-16}[:SELected]:EYE:INPut:PERSistence? 5-145
:CALCulate{1-16}[:SELected]:EYE:INPut:RTIMe:DATA <NRF> 5-140
:CALCulate{1-16}[:SELected]:EYE:INPut:RTIMe:DATA? 5-140
:CALCulate{1-16}[:SELected]:EYE:RESults:DATA? 5-146
:CALCulate{1-16}[:SELected]:FORMat <char> 5-147
:CALCulate{1-16}[:SELected]:FORMat? 5-147
:CALCulate{1-16}[:SELected]:GCOMpression:INDicator[:STATe] <char> 5-149
:CALCulate{1-16}[:SELected]:GCOMpression:INDicator[:STATe]? 5-149
:CALCulate{1-16}[:SELected]:GCOMpression:PARameter <char> 5-149
:CALCulate{1-16}[:SELected]:GCOMpression:PARameter? 5-149
:CALCulate{1-16}[:SELected]:GCOMpression:PVALue <NRf> 5-150
:CALCulate{1-16}[:SELected]:GCOMpression:PVALue? 5-150
:CALCulate{1-16}[:SELected]:GCOMpression:REFerence <char> 5-150
:CALCulate{1-16}[:SELected]:GCOMpression:REFerence:VALue <NRf> 5-151
:CALCulate{1-16}[:SELected]:GCOMpression:REFerence:VALue? 5-151
:CALCulate{1-16}[:SELected]:GCOMpression:REFerence? 5-150
:CALCulate{1-16}[:SELected]:GCOMpression:SNORmalize[:STATe] <char> 5-151
:CALCulate{1-16}[:SELected]:GCOMpression:SNORmalize[:STATe]? 5-151
:CALCulate{1-16}[:SELected]:GDELay:APERture <NRf> 5-152
:CALCulate{1-16}[:SELected]:GDELay:APERture? 5-152
:CALCulate{1-16}[:SELected]:LIMit:DATA <block> 5-154
:CALCulate{1-16}[:SELected]:LIMit:DATA? 5-154
:CALCulate{1-16}[:SELected]:LIMit:DISPlay[:STATe] <char> 5-154
:CALCulate{1-16}[:SELected]:LIMit:DISPlay[:STATe]? 5-154
:CALCulate{1-16}[:SELected]:LIMit:FAIL? 5-154
:CALCulate{1-16}[:SELected]:LIMit:OFF 5-154
:CALCulate{1-16}[:SELected]:LIMit:REPort:POINt? 5-155
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:ADD {No argument} | {<char>} | {<char>,<NRf>,<NRf>} 5-155
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:CLEar 5-156
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:COUNt? 5-156

:CALCulate{1-16}[:SELected]:LIMit:SEGMent:DEFine <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf> 5-156
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:DEFine? 5-156
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:RADius <NRf> 5-157
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:RADius? 5-157
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:TYPe <char> 5-157
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:TYPe? 5-157
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X1 <NRf> 5-158
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X1? 5-158
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X2 <NRf> 5-158
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:X2? 5-158
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y1 <NRf> 5-158
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y1? 5-158
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y12 <NRf> 5-159
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y12? 5-159
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y2 <NRf> 5-159
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y2? 5-159
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y22 <NRf> 5-159
:CALCulate{1-16}[:SELected]:LIMit:SEGMent:Y22? 5-159
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:DEFine <NRf> | <NRf>,<NRf> | <NRf>,<NRf>,<NRf>,<NRf>} 5-160
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:DEFine? 5-160
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:DELete 5-160
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:RADius <NRf> 5-161
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:RADius? 5-161
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:TYPe <char> 5-161
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:TYPe? 5-161
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X1 <NRf> 5-162
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X1? 5-162
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X2 <NRf> 5-162
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:X2? 5-162
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y1 <NRf> 5-162
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y1? 5-162
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y12 <NRf> 5-163
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y12? 5-163
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y2 <NRf> 5-163
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y2? 5-163
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y22 <NRf> 5-164
:CALCulate{1-16}[:SELected]:LIMit:SEGMent{1-50}:Y22? 5-164
:CALCulate{1-16}[:SELected]:LIMit[:STATe] <char> 5-164
:CALCulate{1-16}[:SELected]:LIMit[:STATe]? 5-164
:CALCulate{1-16}[:SELected]:MARKer:ACTivate? 5-165
:CALCulate{1-16}[:SELected]:MARKer:ALL[:STATe] <char> 5-165
:CALCulate{1-16}[:SELected]:MARKer:MOVe:CENTer 5-165
:CALCulate{1-16}[:SELected]:MARKer:MOVe:REFMarker 5-166
:CALCulate{1-16}[:SELected]:MARKer:MOVe:STARt 5-166
:CALCulate{1-16}[:SELected]:MARKer:MOVe:STOP 5-166
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch 5-166
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:EXCursion <NRf> 5-167
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:EXCursion? 5-167
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:POLarity <char> 5-167
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:POLarity? 5-167
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:THREshold <NRf> 5-167
:CALCulate{1-16}[:SELected]:MARKer:MPSEArch:THREshold? 5-167
:CALCulate{1-16}[:SELected]:MARKer:MTSEArch 5-167
:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TARget <NRf> 5-168
:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TARget? 5-168
:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TRANsition <char> 5-168
:CALCulate{1-16}[:SELected]:MARKer:MTSEArch:TRANsition? 5-168
:CALCulate{1-16}[:SELected]:MARKer:OFF 5-168
:CALCulate{1-16}[:SELected]:MARKer:PSEArch <char> 5-168
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:EXCursion <NRf> 5-169
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:EXCursion? 5-169
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:POLarity <char> 5-169
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:POLarity? 5-169
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:THREshold <NRf> 5-169
:CALCulate{1-16}[:SELected]:MARKer:PSEArch:THREshold? 5-169
:CALCulate{1-16}[:SELected]:MARKer:PSEArch? 5-168
:CALCulate{1-16}[:SELected]:MARKer:SEArch <char> 5-170

:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:DATA? 5-170
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:DEFine <NRf> 5-170
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:DEFine? 5-170
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:HIGH <NRf> 5-171
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:HIGH? 5-171
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:LOW <NRf> 5-171
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe:LOW? 5-171
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe[:STATe] <char> 5-171
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth:SHAPe[:STATe]? 5-171
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth[:STATe] <char> 5-172
:CALCulate{1-16}[:SELected]:MARKer:SEArch:BANDwidth[:STATe]? 5-172
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:DATA? 5-172
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:DEFine <NRf> 5-172
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:DEFine? 5-172
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:HIGH <NRf> 5-173
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:HIGH? 5-173
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:LOW <NRf> 5-173
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe:LOW? 5-173
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe[:STATe] <char> 5-173
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh:SHAPe[:STATe]? 5-173
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh[:STATe] <char> 5-174
:CALCulate{1-16}[:SELected]:MARKer:SEArch:NOTCh[:STATe]? 5-174
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:ALL[:STATe] <char> 5-174
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:ALL[:STATe]? 5-174
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STARt:X <NRf> 5-174
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STARt:X? 5-174
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STOP:X <NRf> 5-175
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe:STOP:X? 5-175
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe[:STATe] <char> 5-175
:CALCulate{1-16}[:SELected]:MARKer:SEArch:RANGe[:STATe]? 5-175
:CALCulate{1-16}[:SELected]:MARKer:SEArch:TRACKing[:STATe] <char> 5-175
:CALCulate{1-16}[:SELected]:MARKer:SEArch:TRACKing[:STATe]? 5-175
:CALCulate{1-16}[:SELected]:MARKer:SEArch? 5-170
:CALCulate{1-16}[:SELected]:MARKer:SET:CENTer 5-175
:CALCulate{1-16}[:SELected]:MARKer:SET:REFLevel 5-176
:CALCulate{1-16}[:SELected]:MARKer:SET:STARt 5-176
:CALCulate{1-16}[:SELected]:MARKer:SET:STOP 5-176
:CALCulate{1-16}[:SELected]:MARKer:TSEArch <char> 5-176
:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TARget <NRf> 5-177
:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TARget? 5-177
:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TRANsition <char> 5-177
:CALCulate{1-16}[:SELected]:MARKer:TSEArch:TRANsition? 5-177
:CALCulate{1-16}[:SELected]:MARKer:TSEArch? 5-176
:CALCulate{1-16}[:SELected]:MARKer{1-13}:ACTivate 5-178
:CALCulate{1-16}[:SELected]:MARKer{1-13}:MOVe <char> 5-179
:CALCulate{1-16}[:SELected]:MARKer{1-13}:SET <char> 5-179
:CALCulate{1-16}[:SELected]:MARKer{1-13}:X <NRf> 5-179
:CALCulate{1-16}[:SELected]:MARKer{1-13}:X? 5-179
:CALCulate{1-16}[:SELected]:MARKer{1-13}:Y? 5-180
:CALCulate{1-16}[:SELected]:MARKer{1-13}[:STATe] <char> 5-180
:CALCulate{1-16}[:SELected]:MARKer{1-13}[:STATe]? 5-180
:CALCulate{1-16}[:SELected]:MATH:DATa:FILe:MEMorize 5-181
:CALCulate{1-16}[:SELected]:MATH:DISPlay <char> 5-182
:CALCulate{1-16}[:SELected]:MATH:DISPlay? 5-182
:CALCulate{1-16}[:SELected]:MATH:FUNCtion <char> 5-182
:CALCulate{1-16}[:SELected]:MATH:FUNCtion? 5-182
:CALCulate{1-16}[:SELected]:MATH:INTErtrace:FUNCtion <char> 5-182
:CALCulate{1-16}[:SELected]:MATH:INTErtrace:FUNCtion? 5-182
:CALCulate{1-16}[:SELected]:MATH:INTErtrace:OPERand{1-2}:DEFine <char1>, <char2> 5-183
:CALCulate{1-16}[:SELected]:MATH:INTErtrace:OPERand{1-2}:DEFine? 5-183
:CALCulate{1-16}[:SELected]:MATH:INTErtrace[:STATe] <char> 5-184
:CALCulate{1-16}[:SELected]:MATH:INTErtrace[:STATe]? 5-184
:CALCulate{1-16}[:SELected]:MATH:MEMorize 5-184
:CALCulate{1-16}[:SELected]:MDATA:FDATa <block> 5-185
:CALCulate{1-16}[:SELected]:MDATA:FDATa? 5-185
:CALCulate{1-16}[:SELected]:MDATA:SDATa <block> 5-186
:CALCulate{1-16}[:SELected]:MDATA:SDATa? 5-186

:CALCulate{1-16}[:SELected]:MSTatistics:EYE:AMPLitude[:STATe] <char> 5-187
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:AMPLitude[:STATe]? 5-187
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ARELated[:STATe] <char> 5-187
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ARELated[:STATe]? 5-187
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:CPERcentage[:STATe] <char> 5-188
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:CPERcentage[:STATe]? 5-188
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:DCDistortion[:STATe] <char> 5-188
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:DCDistortion[:STATe]? 5-188
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:FTIMe[:STATe] <char> 5-189
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:FTIMe[:STATe]? 5-189
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:HEIGht[:STATe] <char> 5-189
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:HEIGht[:STATe]? 5-189
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MARKer:POSition <char> 5-189
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MARKer:POSition? 5-189
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MLEVel[:STATe] <char> 5-190
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:MLEVel[:STATe]? 5-190
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OFACtor[:STATe] <char> 5-190
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OFACtor[:STATe]? 5-190
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OLEVel[:STATe] <char> 5-190
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:OLEVel[:STATe]? 5-190
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:PPJitter[:STATe] <char> 5-191
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:PPJitter[:STATe]? 5-191
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RMSJitter[:STATe] <char> 5-191
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RMSJitter[:STATe]? 5-191
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RTIMe[:STATe] <char> 5-191
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:RTIMe[:STATe]? 5-191
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:SNRatio[:STATe] <char> 5-192
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:SNRatio[:STATe]? 5-192
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:TRELated[:STATe] <char> 5-192
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:TRELated[:STATe]? 5-192
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:WIDth[:STATe] <char> 5-192
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:WIDth[:STATe]? 5-192
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ZLEVel[:STATe] <char> 5-193
:CALCulate{1-16}[:SELected]:MSTatistics:EYE:ZLEVel[:STATe]? 5-193
:CALCulate{1-16}[:SELected]:SMITh:IMPedance <NRf> 5-194
:CALCulate{1-16}[:SELected]:SMITh:IMPedance? 5-194
:CALCulate{1-16}[:SELected]:SMITh:WRAP <NRf> 5-195
:CALCulate{1-16}[:SELected]:SMITh:WRAP? 5-195
:CALCulate{1-16}[:SELected]:SMOothing:APERture <NRf> 5-196
:CALCulate{1-16}[:SELected]:SMOothing:APERture? 5-196
:CALCulate{1-16}[:SELected]:SMOothing[:STATe] <char> 5-196
:CALCulate{1-16}[:SELected]:SMOothing[:STATe]? 5-196
:CALCulate{1-16}[:SELected]:TDATA:FDATa <block> 5-197
:CALCulate{1-16}[:SELected]:TDATA:FDATa? 5-197
:CALCulate{1-16}[:SELected]:TDATA:SDATa <block> 5-198
:CALCulate{1-16}[:SELected]:TDATA:SDATa? 5-198
:CALCulate{1-16}[:SELected]:TRANsform:TIME:ALIAsfree? 5-200
:CALCulate{1-16}[:SELected]:TRANsform:TIME:CENTer <NRf> 5-200
:CALCulate{1-16}[:SELected]:TRANsform:TIME:CENTer? 5-200
:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm <char> 5-201
:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm:OTHer <NRf> 5-201
:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm:OTHer? 5-201
:CALCulate{1-16}[:SELected]:TRANsform:TIME:DCTerm? 5-201
:CALCulate{1-16}[:SELected]:TRANsform:TIME:DISTance? 5-201
:CALCulate{1-16}[:SELected]:TRANsform:TIME:EXTrapolate <char> 5-202
:CALCulate{1-16}[:SELected]:TRANsform:TIME:EXTrapolate? 5-202
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:CENTer <NRf> 5-202
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:CENTer? 5-202
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:DCGamma <NRf> 5-202
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:DCGamma? 5-202
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:KBBeta <NRf> 5-203
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:KBBeta? 5-203
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:NOTCh[:STATe] <char> 5-203
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:NOTCh[:STATe]? 5-203
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SHAPe <char> 5-204
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SHAPe? 5-204
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SPAN <NRf> 5-204

:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:SPAN? 5-204
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STARt <NRf> 5-205
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STARt? 5-205
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STOP <NRf> 5-205
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE:STOP? 5-205
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE[:STATe] <char> 5-205
:CALCulate{1-16}[:SELected]:TRANsform:TIME:GATE[:STATe]? 5-205
:CALCulate{1-16}[:SELected]:TRANsform:TIME:IMPUlsewidth? 5-206
:CALCulate{1-16}[:SELected]:TRANsform:TIME:PHASor:IMPulse[:STATe] <char> 5-206
:CALCulate{1-16}[:SELected]:TRANsform:TIME:PHASor:IMPulse[:STATe]? 5-206
:CALCulate{1-16}[:SELected]:TRANsform:TIME:RESPonse <char> 5-206
:CALCulate{1-16}[:SELected]:TRANsform:TIME:RESPonse? 5-206
:CALCulate{1-16}[:SELected]:TRANsform:TIME:SPAN <NRf> 5-207
:CALCulate{1-16}[:SELected]:TRANsform:TIME:SPAN? 5-207
:CALCulate{1-16}[:SELected]:TRANsform:TIME:STARt <NRf> 5-207
:CALCulate{1-16}[:SELected]:TRANsform:TIME:STARt? 5-207
:CALCulate{1-16}[:SELected]:TRANsform:TIME:STOP <NRf> 5-207
:CALCulate{1-16}[:SELected]:TRANsform:TIME:STOP? 5-207
:CALCulate{1-16}[:SELected]:TRANsform:TIME:TIME? 5-208
:CALCulate{1-16}[:SELected]:TRANsform:TIME:TRIP <char> 5-208
:CALCulate{1-16}[:SELected]:TRANsform:TIME:TRIP? 5-208
:CALCulate{1-16}[:SELected]:TRANsform:TIME:TYPe <char> 5-208
:CALCulate{1-16}[:SELected]:TRANsform:TIME:TYPe? 5-208
:CALCulate{1-16}[:SELected]:TRANsform:TIME:UNIt <char> 5-208
:CALCulate{1-16}[:SELected]:TRANsform:TIME:UNIt? 5-208
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:DCGamma <NRf> 5-209
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:DCGamma? 5-209
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:KBBeta <NRf> 5-209
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:KBBeta? 5-209
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:SHAPe <char> 5-210
:CALCulate{1-16}[:SELected]:TRANsform:TIME:WINDow:SHAPe? 5-210
:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM:IMPulses <NRf> 5-210
:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM:IMPulses? 5-210
:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM[:STATe] <char> 5-211
:CALCulate{1-16}[:SELected]:TRANsform:TIME:ZOOM[:STATe]? 5-211
:CONTrol:AUXio:LIMit:POLarity <char> 5-215
:CONTrol:AUXio:LIMit:POLarity? 5-215
:CONTrol:AUXio:LIMit[:STATe] <char> 5-215
:CONTrol:AUXio:LIMit[:STATe]? 5-215
:CONTrol{1-16}:AOUT:MODE <char> 5-218
:CONTrol{1-16}:AOUT:MODE? 5-218
:CONTrol{1-16}:AOUT:PULSe:WIDth <NRf> 5-219
:CONTrol{1-16}:AOUT:PULSe:WIDth? 5-219
:CONTrol{1-16}:AOUT:VERTical:TRACe <char> 5-219
:CONTrol{1-16}:AOUT:VERTical:TRACe:ACTive[:STATe] <char> 5-220
:CONTrol{1-16}:AOUT:VERTical:TRACe:ACTive[:STATe]? 5-220
:CONTrol{1-16}:AOUT:VERTical:TRACe? 5-219
:CONTrol{1-16}:AOUT:VOLTage:STARt <NRf> 5-220
:CONTrol{1-16}:AOUT:VOLTage:STARt? 5-220
:CONTrol{1-16}:AOUT:VOLTage:STOP <NRf> 5-220
:CONTrol{1-16}:AOUT:VOLTage:STOP? 5-220
:CONTrol{1-16}:AOUT:VOLTage:VMAX <NRf> 5-221
:CONTrol{1-16}:AOUT:VOLTage:VMAX? 5-221
:CONTrol{1-16}:AOUT:VOLTage:VMIN <NRf> 5-221
:CONTrol{1-16}:AOUT:VOLTage:VMIN? 5-221
:CONTrol{1-16}:AOUT[:STATe] <char> 5-221
:CONTrol{1-16}:AOUT[:STATe]? 5-221
:CONTrol{1-16}:AOUT{1-2}:DRIVen:LEV <NRf> 5-222
:CONTrol{1-16}:AOUT{1-2}:DRIVen:LEV? 5-222
:CONTrol{1-16}:AOUT{1-2}:TTL:TYPe <char> 5-222
:CONTrol{1-16}:AOUT{1-2}:TTL:TYPe? 5-222
:DISPlay:ACTive:CHANnel:SWEep[:STATe] <char> 5-224
:DISPlay:ACTive:CHANnel:SWEep[:STATe]? 5-224
:DISPlay:COLor:INVert:BACK <NRf>, <NRf>, <NRf> 5-224
:DISPlay:COLor:INVert:BACK? 5-224
:DISPlay:COLor:INVert:GRATicule:MAIN <NRf>, <NRf>, <NRf> 5-224
:DISPlay:COLor:INVert:GRATicule:MAIN? 5-224

:DISPlay:COLor:INVert:GRATicule:SUB <NRf>, <NRf>, <NRf> . . . 5-225
:DISPlay:COLor:INVert:GRATicule:SUB? . . . 5-225
:DISPlay:COLor:INVert:LIMit <NRf>, <NRf>, <NRf> . . . 5-225
:DISPlay:COLor:INVert:LIMit? . . . 5-225
:DISPlay:COLor:INVert:TRACe{1-16}:DATA <NRf>, <NRf>, <NRf> . . . 5-226
:DISPlay:COLor:INVert:TRACe{1-16}:DATA? . . . 5-226
:DISPlay:COLor:INVert:TRACe{1-16}:MEMory <NRf>, <NRf>, <NRf> . . . 5-226
:DISPlay:COLor:INVert:TRACe{1-16}:MEMory? . . . 5-226
:DISPlay:COLor:INVert[:STATe] <char> . . . 5-226
:DISPlay:COLor:INVert[:STATe]? . . . 5-226
:DISPlay:COLor:NORMal:BACK <NRf>, <NRf>, <NRf> . . . 5-227
:DISPlay:COLor:NORMal:BACK? . . . 5-227
:DISPlay:COLor:NORMal:GRATicule:MAIN <NRf>, <NRf>, <NRf> . . . 5-227
:DISPlay:COLor:NORMal:GRATicule:MAIN? . . . 5-227
:DISPlay:COLor:NORMal:GRATicule:SUB <NRf>, <NRf>, <NRf> . . . 5-227
:DISPlay:COLor:NORMal:GRATicule:SUB? . . . 5-227
:DISPlay:COLor:NORMal:LIMit <NRf>, <NRf>, <NRf> . . . 5-228
:DISPlay:COLor:NORMal:LIMit? . . . 5-228
:DISPlay:COLor:NORMal:TRACe{1-16}:DATA <NRf>, <NRf>, <NRf> . . . 5-228
:DISPlay:COLor:NORMal:TRACe{1-16}:DATA? . . . 5-228
:DISPlay:COLor:NORMal:TRACe{1-16}:MEMory <NRf>, <NRf>, <NRf> . . . 5-228
:DISPlay:COLor:NORMal:TRACe{1-16}:MEMory? . . . 5-228
:DISPlay:COLor:RESet . . . 5-229
:DISPlay:COUNt <NRf> . . . 5-229
:DISPlay:COUNt? . . . 5-229
:DISPlay:FSIGn[:STATe] <char> . . . 5-229
:DISPlay:FSIGn[:STATe]? . . . 5-229
:DISPlay:SIZe <char> . . . 5-230
:DISPlay:SIZe? . . . 5-230
:DISPlay:SPLit <char> . . . 5-230
:DISPlay:SPLit? . . . 5-230
:DISPlay:WINDow:ACTivate? . . . 5-231
:DISPlay:WINDow:Y:NDIVisions <NRf> . . . 5-231
:DISPlay:WINDow:Y:NDIVisions? . . . 5-231
:DISPlay:WINDow{1-16}:ACTivate . . . 5-231
:DISPlay:WINDow{1-16}:SIZe <char> . . . 5-232
:DISPlay:WINDow{1-16}:SIZe? . . . 5-232
:DISPlay:WINDow{1-16}:SPLit <char> . . . 5-233
:DISPlay:WINDow{1-16}:SPLit? . . . 5-233
:DISPlay:WINDow{1-16}:TITLe <string> . . . 5-234
:DISPlay:WINDow{1-16}:TITLe:STATe <char> . . . 5-234
:DISPlay:WINDow{1-16}:TITLe:STATe? . . . 5-234
:DISPlay:WINDow{1-16}:TITLe? . . . 5-234
:DISPlay:WINDow{1-16}:TRACe{1-16}:SIZe <char> . . . 5-234
:DISPlay:WINDow{1-16}:TRACe{1-16}:SIZe? . . . 5-234
:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:X <NRf> . . . 5-235
:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:X? . . . 5-235
:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:Y <NRf> . . . 5-235
:DISPlay:WINDow{1-16}:TRACe{1-16}:SMITh:CENTer:Y? . . . 5-235
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:AUTO . . . 5-236
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV <NRf> . . . 5-237
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV? . . . 5-237
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV2 <NRf> . . . 5-238
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PDIV2? . . . 5-238
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:OFFSet <NRf> . . . 5-238
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:OFFSet? . . . 5-238
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:WRAPping[:STATe] <char> . . . 5-238
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHASe:WRAPping[:STATe]? . . . 5-238
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHOFF <NRf> . . . 5-239
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:PHOFF? . . . 5-239
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV <NRf> . . . 5-239
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV? . . . 5-239
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV2 <NRf> . . . 5-239
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RLEV2? . . . 5-239
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS <NRf> . . . 5-240
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS? . . . 5-240
:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS2 <NRf> . . . 5-240

:DISPlay:WINDow{1-16}:TRACe{1-16}:Y:RPOS2? 5-240
:DISPlay:WINDow{1-16}:Y:AUTO 5-240
:DISPlay:Y:AUTO 5-241
:DISPlay:Y:NDIVisions 5-241
:FORMat:BORDer <char> 5-242
:FORMat:BORDer? 5-242
:FORMat:DATA <char> 5-243
:FORMat:DATA:HEADing[:STATe] <char> 5-243
:FORMat:DATA:HEADing[:STATe]? 5-243
:FORMat:DATA? 5-243
:FORMat:SNP:FREQuency <char> 5-243
:FORMat:SNP:FREQuency? 5-243
:FORMat:SNP:PARameter <char> 5-244
:FORMat:SNP:PARameter? 5-244
:HCOPy:DEVice:ID <String> 5-245
:HCOPy:DEVice:ID:STATe <char> 5-245
:HCOPy:DEVice:ID:STATe? 5-245
:HCOPy:DEVice:ID? 5-245
:HCOPy:IMAGe <char> 5-246
:HCOPy:IMAGe? 5-246
:HCOPy:MODel <String> 5-246
:HCOPy:MODel:STATe? 5-246
:HCOPy:MODel:STATe<char> 5-246
:HCOPy:MODel? 5-246
:HCOPy:OPERator:COMMent:STATe <char> 5-247
:HCOPy:OPERator:COMMent:STATe? 5-247
:HCOPy:OPERator:COMMent? 5-247
:HCOPy:OPERator:COMMent<String> 5-247
:HCOPy:OPERator:NAMe <String> 5-247
:HCOPy:OPERator:NAMe:STATe <char> 5-248
:HCOPy:OPERator:NAMe:STATe? 5-248
:HCOPy:OPERator:NAMe? 5-247
:HCOPy:PRINt:DATe:TIMe:STATe <char> 5-248
:HCOPy:PRINt:DATe:TIMe:STATe? 5-248
:HCOPy:PRINt:HEADers:STATe <char> 5-248
:HCOPy:PRINt:HEADers:STATe? 5-248
:HCOPy:PRINt:LOGO:STATe <char> 5-249
:HCOPy:PRINt:LOGO:STATe? 5-249
:HCOPy:PRINt:LOGO:TYPe <char> 5-249
:HCOPy:PRINt:LOGO:TYPe? 5-249
:HCOPy:PRINt:TYPe <char> 5-249
:HCOPy:PRINt:TYPe? 5-249
:HCOPy:USER:LOGO 5-250
:HCOPy:USER:LOGO? 5-250
:HCOPy[:IMMediate] 5-250
:MMEMory:CATalog? {<string>} 5-251
:MMEMory:COPY <string1>, <string2> 5-251
:MMEMory:DELete <string> 5-251
:MMEMory:LOAD <string> 5-252
:MMEMory:LOAD:CKIT <string> 5-252
:MMEMory:LOAD:FLAT <string> 5-252
:MMEMory:LOAD:FSEGMent <string> 5-253
:MMEMory:LOAD:ISEGMent <string> 5-253
:MMEMory:LOAD:LIMit <string> 5-253
:MMEMory:LOAD:LINearity <string> 5-253
:MMEMory:LOAD:MDATA <string> 5-253
:MMEMory:MDIRectory <string> 5-254
:MMEMory:RDIRectory <string> 5-254
:MMEMory:STORe <string> 5-254
:MMEMory:STORe:FLAT{1-7} <string> 5-254
:MMEMory:STORe:FSEGMent <string> 5-255
:MMEMory:STORe:IMAGe <string> 5-255
:MMEMory:STORe:ISEGMent <string> 5-255
:MMEMory:STORe:LIMit <string> 5-255
:MMEMory:STORe:LINearity{1-2} <string> 5-256
:MMEMory:STORe:MDATA <string> 5-256
:MMEMory:STORe:MP41 <string> 5-256

:MMEMory:STORe:MP42 <string> . . . 5-256
:MMEMory:TRANsfer <string>, <block> . . . 5-257
:MMEMory:TRANsfer? . . . 5-257
:MMEMory:WRITE:CKIT <char1>, <char2>, <char3>, <string> . . . 5-257
:SENSe:BANDwidth:ENHancer[:STATe] <char> . . . 5-260
:SENSe:BANDwidth:ENHancer[:STATe]? . . . 5-260
:SENSe:HOLD:FUNCtion <char> . . . 5-261
:SENSe:HOLD:FUNCtion? . . . 5-261
:SENSe{1-16}:ABORtcal . . . 5-264
:SENSe{1-16}:AVERage:CLEar . . . 5-265
:SENSe{1-16}:AVERage:COUNt <NRf> . . . 5-265
:SENSe{1-16}:AVERage:COUNt? . . . 5-265
:SENSe{1-16}:AVERage:SWEep? . . . 5-265
:SENSe{1-16}:AVERage:TYPe <char> . . . 5-266
:SENSe{1-16}:AVERage:TYPe? . . . 5-266
:SENSe{1-16}:AVERage[:STATe] <char> . . . 5-266
:SENSe{1-16}:AVERage[:STATe]? . . . 5-266
:SENSe{1-16}:BANDwidth[:RESolution] <NRf> . . . 5-267
:SENSe{1-16}:BANDwidth[:RESolution]? . . . 5-267
:SENSe{1-16}:BWIDth[:RESolution] <NRf> . . . 5-268
:SENSe{1-16}:BWIDth[:RESolution]? . . . 5-268
:SENSe{1-16}:CORRection:COEFficient <char>, <block> . . . 5-270
:SENSe{1-16}:CORRection:COEFficient:1P2PF . . . 5-275
:SENSe{1-16}:CORRection:COEFficient:1P2PR . . . 5-275
:SENSe{1-16}:CORRection:COEFficient:FULL1 . . . 5-275
:SENSe{1-16}:CORRection:COEFficient:FULL2 . . . 5-276
:SENSe{1-16}:CORRection:COEFficient:FULL4 . . . 5-270
:SENSe{1-16}:CORRection:COEFficient:FULLB . . . 5-276
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PF . . . 5-271
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PR . . . 5-271
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2 . . . 5-271
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULLB . . . 5-272
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:RESPB . . . 5-272
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRB . . . 5-272
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRF . . . 5-272
:SENSe{1-16}:CORRection:COEFficient:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRR . . . 5-273
:SENSe{1-16}:CORRection:COEFficient:PORT{123 | 124 | 134 | 234}:FULL3 . . . 5-273
:SENSe{1-16}:CORRection:COEFficient:PORT{1-4}:FULL1 . . . 5-273
:SENSe{1-16}:CORRection:COEFficient:PORT{1-4}:RESP1 . . . 5-273
:SENSe{1-16}:CORRection:COEFficient:RESP1 . . . 5-276
:SENSe{1-16}:CORRection:COEFficient:RESPB . . . 5-277
:SENSe{1-16}:CORRection:COEFficient:TFRB . . . 5-277
:SENSe{1-16}:CORRection:COEFficient:TFRF . . . 5-277
:SENSe{1-16}:CORRection:COEFficient:TFRR . . . 5-278
:SENSe{1-16}:CORRection:COEFficient? <char> . . . 5-270
:SENSe{1-16}:CORRection:COLLect:1P2PF . . . 5-383
:SENSe{1-16}:CORRection:COLLect:1P2PR . . . 5-383
:SENSe{1-16}:CORRection:COLLect:CALB:1P2PF . . . 5-279
:SENSe{1-16}:CORRection:COLLect:CALB:1P2PR . . . 5-279
:SENSe{1-16}:CORRection:COLLect:CALB:FULL2 . . . 5-280
:SENSe{1-16}:CORRection:COLLect:DISPlay[:CURRent]:CALibration[:STATe] <char> . . . 5-287
:SENSe{1-16}:CORRection:COLLect:DISPlay[:CURRent]:CALibration[:STATe]? . . . 5-287
:SENSe{1-16}:CORRection:COLLect:ECAL:AUTOmatic:ORIentation[:STATe] <char> . . . 5-281
:SENSe{1-16}:CORRection:COLLect:ECAL:AUTOmatic:ORIentation[:STATe]? . . . 5-281
:SENSe{1-16}:CORRection:COLLect:ECAL:BEGin? . . . 5-281
:SENSe{1-16}:CORRection:COLLect:ECAL:CALB:TRUEthru <char> . . . 5-282
:SENSe{1-16}:CORRection:COLLect:ECAL:CHARacterize? . . . 5-282
:SENSe{1-16}:CORRection:COLLect:ECAL:CONTinue? . . . 5-283
:SENSe{1-16}:CORRection:COLLect:ECAL:MSGS:LIST? . . . 2-55
:SENSe{1-16}:CORRection:COLLect:ECAL:MSGS:LIST? . . . 5-285
:SENSe{1-16}:CORRection:COLLect:ECAL:MSGS:LIST? . . . 5-383
:SENSe{1-16}:CORRection:COLLect:ECAL:ORIentation <char> . . . 5-286
:SENSe{1-16}:CORRection:COLLect:ECAL:ORIentation? . . . 5-286
:SENSe{1-16}:CORRection:COLLect:ECAL:PORT{12|13|14|23|24|34}:FULL4 . . . 5-286
:SENSe{1-16}:CORRection:COLLect:ECAL[:CALa]:TRUEthru <char> . . . 5-286
:SENSe{1-16}:CORRection:COLLect:ECAL[:CALa]:TRUEthru? . . . 5-286
:SENSe{1-16}:CORRection:COLLect:ENHMatch:MIXer:USE:TSM[:STATe] . . . 5-384

:SENSe{1-16}:CORRection:COLLect:ENHMatch:MIXer:USE:TSM[:STATe]? 5-384
:SENSe{1-16}:CORRection:COLLect:FLEXible:DEFine <char>{,<char2>,...,<charn>} 5-288
:SENSe{1-16}:CORRection:COLLect:FLEXible:DEFine? 5-288
:SENSe{1-16}:CORRection:COLLect:FLEXible[:STATe] <char> 5-288
:SENSe{1-16}:CORRection:COLLect:FLEXible[:STATe]? 5-288
:SENSe{1-16}:CORRection:COLLect:FULL1 5-385
:SENSe{1-16}:CORRection:COLLect:FULL2 5-385
:SENSe{1-16}:CORRection:COLLect:FULL4 5-289
:SENSe{1-16}:CORRection:COLLect:FULLB 5-385
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:CAL1:FILename 5-386
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:CAL2:FILename 5-386
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:REFPlane:EXTension :MODel 5-386
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:REFPlane:EXTension :MODel? 5-386
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:S2P:FILename 5-386
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:S2P:REVerse[:STATe] 5-387
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:S2P:REVerse[:STATe]? 5-387
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:DIELectric:TYPe 5-387
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:DIELectric:TYPe? 5-387
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:FREQuency 5-387
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:FREQuency? 5-387
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:IMPedance 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:IMPedance? 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LENGth 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LENGth? 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LOSS 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:LOSS? 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:OTHer 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:ENHMatch:TLINe:OTHer? 5-388
:SENSe{1-16}:CORRection:COLLect:HYBRid:FILe{1-4} <string> 5-290
:SENSe{1-16}:CORRection:COLLect:HYBRid:FILe{1-4}? 5-290
:SENSe{1-16}:CORRection:COLLect:HYBRid:FULL4 5-290
:SENSe{1-16}:CORRection:COLLect:HYBRid:MULTiple:THRu <char> {,<char>,...,<char>} 5-291
:SENSe{1-16}:CORRection:COLLect:HYBRid:MULTiple:THRu? 5-291
:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2 5-291
:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL4 5-293
:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu 5-293
:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal[:STATe] <char> 5-294
:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal[:STATe]? 5-294
:SENSe{1-16}:CORRection:COLLect:HYBRid:PORT{123 | 124 | 134 | 234}:FULL3 5-292
:SENSe{1-16}:CORRection:COLLect:LINE <char> 5-389
:SENSe{1-16}:CORRection:COLLect:LINE? 5-389
:SENSe{1-16}:CORRection:COLLect:LOAD <char> 5-389
:SENSe{1-16}:CORRection:COLLect:LOAD? 5-389
:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:LOAD <string> 5-318
:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:NAMe <char> 5-318
:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:NAMe? 5-318
:SENSe{1-16}:CORRection:COLLect:LRL:[CALa]:CKIT:SAVe <string> 5-319
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND:COUNt <NRf> 5-295
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND:COUNt? 5-295
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND{1-5}:REFLection:TYPe <char> 5-295
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:BAND{1-5}:REFLection:TYPe? 5-295
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:LOAD <string> 5-296
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:NAMe <char> 5-296
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:NAMe? 5-296
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:CKIT:SAVe <string> 5-297
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE 5-297
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:FREQuency <NRf> 5-297
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:FREQuency? 5-297
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LENGth <NRf> 5-297
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LENGth? 5-297
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LOSS <NRf> 5-298
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:LINE:LOSS? 5-298
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C0 <NRf> 5-298
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C0? 5-298
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C1 <NRf> 5-298
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C1? 5-298
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C2 <NRf> 5-299

:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C2? . . . 5-299
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C3 <NRf> . . . 5-299
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:C3? . . . 5-299
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L0 <NRf> . . . 5-299
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L0? . . . 5-299
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L1 <NRf> . . . 5-300
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L1? . . . 5-300
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L2 <NRf> . . . 5-300
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L2? . . . 5-300
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L3 <NRf> . . . 5-300
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:L3? . . . 5-300
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF1 <NRf> . . . 5-301
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF1? . . . 5-301
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF2 <NRf> . . . 5-301
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF2? . . . 5-301
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF3 <NRf> . . . 5-301
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFF3? . . . 5-301
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFFS <NRf> . . . 5-302
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:OFFS? . . . 5-302
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:R <NRf> . . . 5-302
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:R? . . . 5-302
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:Z0 <NRf> . . . 5-302
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:PORT{1-4}:MATCH:Z0? . . . 5-302
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:TYPe <char> . . . 5-303
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:DEVice{1-10}:TYPe? . . . 5-303
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint <NRf> . . . 5-303
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint? . . . 5-303
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint{1-4} <NRf> . . . 5-303
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:FREQuency:BREakpoint{1-4}? . . . 5-303
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:OPEN:OFFS <NRf> . . . 5-304
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:OPEN:OFFS? . . . 5-304
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:REFPlane <char> . . . 5-304
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:REFPlane? . . . 5-304
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:SHORT:OFFS <NRf> . . . 5-304
:SENSe{1-16}:CORRection:COLLect:LRL:CALB:SHORT:OFFS? . . . 5-304
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:MATCH:PORT <char> . . . 5-305
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:MATCH:PORT? . . . 5-305
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT{1-4}:MATCH . . . 5-305
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE . . . 5-306
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:DELay <NRf> . . . 5-306
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:DELay? . . . 5-306
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:FREQuency <NRf> . . . 5-306
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:FREQuency? . . . 5-306
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LENGth <NRf> . . . 5-307
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LENGth? . . . 5-307
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LOSS <NRf> . . . 5-307
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:LOSS? . . . 5-307
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:PLENgth <NRf> . . . 5-308
:SENSe{1-16}:CORRection:COLLect:LRL:DEVice{1-10}:PORT12:LINE:PLENgth? . . . 5-308
:SENSe{1-16}:CORRection:COLLect:LRL:PORT{13 | 14 | 23 | 24}:FULL3 <char> . . . 5-309
:SENSe{1-16}:CORRection:COLLect:LRL:PORT{13 | 14 | 23 | 24}:FULL4 . . . 5-310
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:LOAD <string> . . . 5-311
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:NAMe <char> . . . 5-311
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:NAMe? . . . 5-311
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:CKIT:SAVe <string> . . . 5-312
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C0 <NRf> . . . 5-312
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C0? . . . 5-312
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C1 <NRf> . . . 5-312
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C1? . . . 5-312
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C2 <NRf> . . . 5-313
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C2? . . . 5-313
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C3 <NRf> . . . 5-313
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:C3? . . . 5-313
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:OFFSet <NRf> . . . 5-313
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:OPEN:OFFSet? . . . 5-313
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:REFLection:TYPe <char> . . . 5-314
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:REFLection:TYPe? . . . 5-314

:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L0 <NRf> 5-314
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L0? 5-314
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L1 <NRf> 5-314
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L1? 5-314
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L2 <NRf> 5-315
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L2? 5-315
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L3 <NRf> 5-315
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:L3? 5-315
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:OFFSet <NRf> 5-315
:SENSe{1-16}:CORRection:COLLect:LRL:SINGleton:SHORt:OFFSet? 5-315
:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:DIELectric <NRf> 5-316
:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:DIELectric? 5-316
:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:FREQuency <NRf> 5-316
:SENSe{1-16}:CORRection:COLLect:LRL:WAVeguide:FREQuency? 5-316
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND:COUNt <NRf> 5-317
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND:COUNt? 5-317
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND{1-5}:REFLection:TYPe <char> 5-317
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:BAND{1-5}:REFLection:TYPe? 5-317
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE 5-319
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:FREQuency <NRf> 5-319
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:FREQuency? 5-319
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LENGth <NRf> 5-320
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LENGth? 5-320
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LOSS <NRf> 5-320
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:LINE:LOSS? 5-320
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C0 <NRf> 5-320
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C0? 5-320
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C1 <NRf> 5-321
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C1? 5-321
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C2 <NRf> 5-321
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C2? 5-321
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C3 <NRf> 5-322
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:C3? 5-322
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L0 <NRf> 5-322
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L0? 5-322
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L1 <NRf> 5-323
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L1? 5-323
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L2 <NRf> 5-323
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L2? 5-323
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L3 <NRf> 5-324
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:L3? 5-324
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF1 <NRf> 5-324
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF1? 5-324
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF2 <NRf> 5-325
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF2? 5-325
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF3 <NRf> 5-325
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFF3? 5-325
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFFS <NRf> 5-326
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:OFFS? 5-326
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:R <NRf> 5-326
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:R? 5-326
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:Z0 <NRf> 5-326
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:PORT{1-4}:MATCH:Z0? 5-326
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:TYPe <char> 5-327
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:DEVice{1-10}:TYPe? 5-327
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint <NRf> 5-327
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint? 5-327
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint{1-4} <NRf> 5-327
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:FREQuency:BREakpoint{1-4}? 5-327
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:OPEN:OFFS <NRf> 5-328
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:OPEN:OFFS? 5-328
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:REFPlane <char> 5-328
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:REFPlane? 5-328
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:SHORT:OFFS <NRf> 5-328
:SENSe{1-16}:CORRection:COLLect:LRL[:CALa]:SHORT:OFFS? 5-328
:SENSe{1-16}:CORRection:COLLect:METHod <char> 5-329
:SENSe{1-16}:CORRection:COLLect:METHod? 5-329

:SENSe{1-16}:CORRection:COLLect:MICrostrip:DIELectric <NRf> ... 5-330
:SENSe{1-16}:CORRection:COLLect:MICrostrip:DIELectric? ... 5-330
:SENSe{1-16}:CORRection:COLLect:MICrostrip:EFFective <NRf> ... 5-331
:SENSe{1-16}:CORRection:COLLect:MICrostrip:EFFective? ... 5-331
:SENSe{1-16}:CORRection:COLLect:MICrostrip:KIT <char> ... 5-332
:SENSe{1-16}:CORRection:COLLect:MICrostrip:KIT? ... 5-332
:SENSe{1-16}:CORRection:COLLect:MICrostrip:PORT{1-4}:CONNector <char> ... 5-333
:SENSe{1-16}:CORRection:COLLect:MICrostrip:PORT{1-4}:CONNector? ... 5-333
:SENSe{1-16}:CORRection:COLLect:MICrostrip:THICKness <NRf> ... 5-333
:SENSe{1-16}:CORRection:COLLect:MICrostrip:THICKness? ... 5-333
:SENSe{1-16}:CORRection:COLLect:MICrostrip:WIDth <NRf> ... 5-334
:SENSe{1-16}:CORRection:COLLect:MICrostrip:WIDth? ... 5-334
:SENSe{1-16}:CORRection:COLLect:MICrostrip:Z0 <NRf> ... 5-334
:SENSe{1-16}:CORRection:COLLect:MICrostrip:Z0? ... 5-334
:SENSe{1-16}:CORRection:COLLect:MULTiple:THRu <char> {<char>,<char>,<char>,<char>,<char>} ... 5-335
:SENSe{1-16}:CORRection:COLLect:MULTiple:THRu? ... 5-335
:SENSe{1-16}:CORRection:COLLect:PORT <char> ... 5-390
:SENSe{1-16}:CORRection:COLLect:PORT? ... 5-390
:SENSe{1-16}:CORRection:COLLect:PORT{1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34 | 123 | 124 | 134 | 234 | 1234}:FULL1 ... 5-337
:SENSe{1-16}:CORRection:COLLect:PORT{1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34 | 123 | 124 | 134 | 234 | 1234}:RESP1 ... 5-337
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:ISOL ... 5-338
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRB ... 5-338
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRF ... 5-338
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:TFRR ... 5-338
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu ... 5-339
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:FREQuency <NRf> ... 5-339
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:FREQuency? ... 5-339
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LABEL <string> ... 5-339
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LABEL? ... 5-339
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LENGth <NRf> ... 5-340
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LENGth? ... 5-340
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LOSS <NRf> ... 5-340
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:LOSS? ... 5-340
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal <char> ... 5-340
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:RECIProcal? ... 5-340
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:SERial <string> ... 5-341
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:SERial? ... 5-341
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:Z0 <NRf> ... 5-341
:SENSe{1-16}:CORRection:COLLect:PORT{12 | 13 | 14 | 23 | 24 | 34}:THRu:Z0? ... 5-341
:SENSe{1-16}:CORRection:COLLect:PORT{123 | 124 | 134 | 234}:FULL3 ... 5-341
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:CONNector <char> ... 5-342
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:CONNector? ... 5-342
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD ... 5-343
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:SELect <char> ... 5-343
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:SELect? ... 5-343
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:TYPe <char> ... 5-344
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD:TYPe? ... 5-344
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C0 <NRf> ... 5-344
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C0? ... 5-344
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C1 <NRf> ... 5-344
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C1? ... 5-344
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C2 <NRf> ... 5-345
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C2? ... 5-345
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C3 <NRf> ... 5-345
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:C3? ... 5-345
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L0 <NRf> ... 5-345
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L0? ... 5-345
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L1 <NRf> ... 5-346
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L1? ... 5-346
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L2 <NRf> ... 5-346
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L2? ... 5-346
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L3 <NRf> ... 5-346
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:L3? ... 5-346
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:LABEL <string> ... 5-347
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:LABEL? ... 5-347
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF1 <NRf> ... 5-347
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF1? ... 5-347

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF2 <NRf> 5-347
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF2? 5-347
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF3 <NRf> 5-348
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFF3? 5-348
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFFS <NRf> 5-348
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:OFFS? 5-348
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:R <NRf> 5-348
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:R? 5-348
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:SERial <string> 5-349
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:SERial? 5-349
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:Z0 <NRf> 5-349
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD1:Z0? 5-349
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C0 <NRf> 5-349
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C0? 5-349
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C1 <NRf> 5-350
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C1? 5-350
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C2 <NRf> 5-350
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C2? 5-350
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C3 <NRf> 5-350
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:C3? 5-350
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L0 <NRf> 5-351
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L0? 5-351
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L1 <NRf> 5-351
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L1? 5-351
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L2 <NRf> 5-351
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L2? 5-351
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L3 <NRf> 5-352
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:L3? 5-352
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:LABEL <string> 5-352
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:LABEL? 5-352
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF1 <NRf> 5-352
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF1? 5-352
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF2 <NRf> 5-353
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF2? 5-353
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF3 <NRf> 5-353
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFF3? 5-353
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFFS <NRf> 5-353
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:OFFS? 5-353
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:R <NRf> 5-354
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:R? 5-354
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:SERial <string> 5-354
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:SERial? 5-354
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:Z0 <NRf> 5-354
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:LOAD2:Z0? 5-354
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN 5-355
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C0 <NRf> 5-355
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C0? 5-355
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C1 <NRf> 5-355
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C1? 5-355
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C2 <NRf> 5-356
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C2? 5-356
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C3 <NRf> 5-356
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:C3? 5-356
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:LABEL <string> 5-356
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:LABEL? 5-356
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:OFFS <NRf> 5-357
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:OFFS? 5-357
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:SERial <string> 5-357
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:OPEN:SERial? 5-357
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:REFLection:COMPonent <char> 5-358
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:REFLection:COMPonent? 5-358
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT 5-358
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L0 <NRf> 5-358
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L0? 5-358
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L1 <NRf> 5-359
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L1? 5-359
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L2 <NRf> 5-359

:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L2? 5-359
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L3 <NRf> 5-359
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:L3? 5-359
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:LABEL <string> 5-360
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:LABEL? 5-360
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:OFFS <NRf> 5-360
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:OFFS? 5-360
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:SERial <string> 5-360
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT:SERial? 5-360
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1 5-361
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L0 <NRf> 5-361
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L0? 5-361
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L1 <NRf> 5-361
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L1? 5-361
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L2 <NRf> 5-362
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L2? 5-362
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L3 <NRf> 5-362
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:L3? 5-362
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:LABEL <string> 5-362
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:LABEL? 5-362
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:OFFS <NRf> 5-363
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:OFFS? 5-363
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:SERial <string> 5-363
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT1:SERial? 5-363
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2 5-363
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L0 <NRf> 5-364
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L0? 5-364
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L1 <NRf> 5-364
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L1? 5-364
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L2 <NRf> 5-364
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L2? 5-364
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L3 <NRf> 5-365
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:L3? 5-365
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:LABEL <string> 5-365
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:LABEL? 5-365
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:OFFS <NRf> 5-365
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:OFFS? 5-365
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:SERial <string> 5-366
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT2:SERial? 5-366
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3 5-366
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L0 <NRf> 5-366
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L0? 5-366
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L1 <NRf> 5-367
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L1? 5-367
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L2 <NRf> 5-367
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L2? 5-367
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L3 <NRf> 5-367
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:L3? 5-367
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:LABEL <string> 5-368
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:LABEL? 5-368
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:OFFS <NRf> 5-368
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:OFFS? 5-368
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:SERial <string> 5-368
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SHORT3:SERial? 5-368
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD1 5-369
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD2 5-369
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD3 5-369
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD4 5-369
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD5 5-369
:SENSe{1-16}:CORRection:COLLect:PORT{1-4}:SLOAD6 5-370
:SENSe{1-16}:CORRection:COLLect:REFerence:Z0 <NRf> 5-371
:SENSe{1-16}:CORRection:COLLect:REFerence:Z0? 5-371
:SENSe{1-16}:CORRection:COLLect:RESP1 5-391
:SENSe{1-16}:CORRection:COLLect:RESPB 5-391
:SENSe{1-16}:CORRection:COLLect:SAVe 5-371
:SENSe{1-16}:CORRection:COLLect:SMCorrection[:STATe] <char> 5-372
:SENSe{1-16}:CORRection:COLLect:SMCorrection[:STATe]? 5-372

:SENSe{1-16}:CORRection:COLLect:TFR:CLEar 5-372
:SENSe{1-16}:CORRection:COLLect:TFRB 5-391
:SENSe{1-16}:CORRection:COLLect:TFRF 5-392
:SENSe{1-16}:CORRection:COLLect:TFRR 5-392
:SENSe{1-16}:CORRection:COLLect:THRu:ADD <char> 5-372
:SENSe{1-16}:CORRection:COLLect:THRu:CLEar 5-372
:SENSe{1-16}:CORRection:COLLect:TYPe? 5-392
:SENSe{1-16}:CORRection:COLLect:WAVeguide:DIELectric <NRf> 5-373
:SENSe{1-16}:CORRection:COLLect:WAVeguide:DIELectric? 5-373
:SENSe{1-16}:CORRection:COLLect:WAVeguide:FREQuency <NRf> 5-373
:SENSe{1-16}:CORRection:COLLect:WAVeguide:FREQuency? 5-373
:SENSe{1-16}:CORRection:COLLect:WAVeguide:KIT <char> 5-374
:SENSe{1-16}:CORRection:COLLect:WAVeguide:KIT? 5-374
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LABel <string> 5-374
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LABel? 5-374
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:L0 <NRf> 5-375
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:L0? 5-375
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:OFFSet <NRf> 5-375
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:OFFSet? 5-375
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:R <NRf> 5-375
:SENSe{1-16}:CORRection:COLLect:WAVeguide:LOAD:R? 5-375
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C0 <NRf> 5-376
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C0? 5-376
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C1 <NRf> 5-376
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C1? 5-376
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C2 <NRf> 5-376
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C2? 5-376
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C3 <NRf> 5-377
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:C3? 5-377
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:OFFSet <NRf> 5-377
:SENSe{1-16}:CORRection:COLLect:WAVeguide:OPEN:OFFSet? 5-377
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SERial <string> 5-377
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SERial? 5-377
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT1:OFFSet <NRf> 5-378
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT1:OFFSet? 5-378
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT2:OFFSet <NRf> 5-378
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT2:OFFSet? 5-378
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT3:OFFSet <NRf> 5-378
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SHORT3:OFFSet? 5-378
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SLOAD:MINF <NRf> 5-379
:SENSe{1-16}:CORRection:COLLect:WAVeguide:SLOAD:MINF? 5-379
:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PF 5-381
:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:1P2PR 5-381
:SENSe{1-16}:CORRection:COLLect[:CALa]:PORT{12 | 13 | 14 | 23 | 24 | 34}:FULL2 5-381
:SENSe{1-16}:CORRection:DISPlay:RSWeep[:STATe] <char> 5-394
:SENSe{1-16}:CORRection:DISPlay:RSWeep[:STATe]? 5-394
:SENSe{1-16}:CORRection:EXTension:PORT{1-4} <NRf> 5-395
:SENSe{1-16}:CORRection:EXTension:PORT{1-4}? 5-395
:SENSe{1-16}:CORRection:INTerpolation:STATe ? 5-396
:SENSe{1-16}:CORRection:INTerpolation:STATe <char> 5-396
:SENSe{1-16}:CORRection:ISOLation:STATe <char> 5-397
:SENSe{1-16}:CORRection:ISOLation:STATe? 5-397
:SENSe{1-16}:CORRection:STATe <char> 5-398
:SENSe{1-16}:CORRection:STATe? 5-398
:SENSe{1-16}:FREQuency:CENTer <NRf> 5-399
:SENSe{1-16}:FREQuency:CENTer? 5-399
:SENSe{1-16}:FREQuency:CW <NRf> 5-399
:SENSe{1-16}:FREQuency:CW? 5-399
:SENSe{1-16}:FREQuency:DATA <block> 5-400
:SENSe{1-16}:FREQuency:DATA? 5-400
:SENSe{1-16}:FREQuency:SPAN <NRf> 5-400
:SENSe{1-16}:FREQuency:SPAN? 5-400
:SENSe{1-16}:FREQuency:STARt <NRf> 5-400
:SENSe{1-16}:FREQuency:STARt? 5-400
:SENSe{1-16}:FREQuency:STOP <NRf> 5-400
:SENSe{1-16}:FREQuency:STOP? 5-400
:SENSe{1-16}:FSEGMent:ADD 5-401

:SENSe{1-16}:FSEGMent:AVERage:COUNt <NRf> 5-402
:SENSe{1-16}:FSEGMent:AVERage:COUNt? 5-402
:SENSe{1-16}:FSEGMent:BWIDth[:RESolution] <NRf> 5-402
:SENSe{1-16}:FSEGMent:BWIDth[:RESolution]? 5-402
:SENSe{1-16}:FSEGMent:CLEar 5-402
:SENSe{1-16}:FSEGMent:COUNt? 5-402
:SENSe{1-16}:FSEGMent:CWMODe[:STATe] <char> 5-403
:SENSe{1-16}:FSEGMent:CWMODe[:STATe]? 5-403
:SENSe{1-16}:FSEGMent:DATA? 5-403
:SENSe{1-16}:FSEGMent:DISPlay <char> 5-403
:SENSe{1-16}:FSEGMent:DISPlay:AVERaging[:STATe] <char> 5-403
:SENSe{1-16}:FSEGMent:DISPlay:AVERaging[:STATe]? 5-403
:SENSe{1-16}:FSEGMent:DISPlay:IFBW[:STATe] <char> 5-404
:SENSe{1-16}:FSEGMent:DISPlay:IFBW[:STATe]? 5-404
:SENSe{1-16}:FSEGMent:DISPlay:POWer[:STATe] <char> 5-404
:SENSe{1-16}:FSEGMent:DISPlay:POWer[:STATe]? 5-404
:SENSe{1-16}:FSEGMent:DISPlay? 5-403
:SENSe{1-16}:FSEGMent:FREQuency:ACTive:STARt? 5-404
:SENSe{1-16}:FSEGMent:FREQuency:ACTive:STOP? 5-405
:SENSe{1-16}:FSEGMent:FREQuency:FSTArt <NRf> 5-405
:SENSe{1-16}:FSEGMent:FREQuency:FSTArt? 5-405
:SENSe{1-16}:FSEGMent:FREQuency:FSTEp <NRf> 5-406
:SENSe{1-16}:FSEGMent:FREQuency:FSTEp? 5-406
:SENSe{1-16}:FSEGMent:FREQuency:FSTOp <NRf> 5-406
:SENSe{1-16}:FSEGMent:FREQuency:FSTOp? 5-406
:SENSe{1-16}:FSEGMent:FREQuency:STARt <NRf> 5-407
:SENSe{1-16}:FSEGMent:FREQuency:STARt? 5-407
:SENSe{1-16}:FSEGMent:FREQuency:STOP <NRf> 5-407
:SENSe{1-16}:FSEGMent:FREQuency:STOP? 5-407
:SENSe{1-16}:FSEGMent:FREQuency[:CW][:FIXed] <NRf> 5-408
:SENSe{1-16}:FSEGMent:FREQuency[:CW][:FIXed]? 5-408
:SENSe{1-16}:FSEGMent:MAXPoints? 5-408
:SENSe{1-16}:FSEGMent:MODBB:POWer:PORT{1-4} <NRf> 5-409
:SENSe{1-16}:FSEGMent:MODBB:POWer:PORT{1-4}? 5-409
:SENSe{1-16}:FSEGMent:POINt? 5-409
:SENSe{1-16}:FSEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude] <NRf> 5-410
:SENSe{1-16}:FSEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]? 5-410
:SENSe{1-16}:FSEGMent:SPAntype <char> 5-410
:SENSe{1-16}:FSEGMent:SPAntype? 5-410
:SENSe{1-16}:FSEGMent:SWEep:MAXimize 5-411
:SENSe{1-16}:FSEGMent:SWEep:POINt <NR1> 5-411
:SENSe{1-16}:FSEGMent:SWEep:POINt? 5-411
:SENSe{1-16}:FSEGMent[:STATe] <char> 5-411
:SENSe{1-16}:FSEGMent[:STATe]? 5-411
:SENSe{1-16}:FSEGMent{1-50}:AVERage:COUNt <NRf> 5-412
:SENSe{1-16}:FSEGMent{1-50}:AVERage:COUNt? 5-412
:SENSe{1-16}:FSEGMent{1-50}:BWIDth[:RESolution] <NRf> 5-412
:SENSe{1-16}:FSEGMent{1-50}:BWIDth[:RESolution]? 5-412
:SENSe{1-16}:FSEGMent{1-50}:CWMODe[:STATe] <char> 5-413
:SENSe{1-16}:FSEGMent{1-50}:CWMODe[:STATe]? 5-413
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTArt <NRf> 5-413
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTArt? 5-413
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTEp <NRf> 5-414
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTEp? 5-414
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTOp <NRf> 5-414
:SENSe{1-16}:FSEGMent{1-50}:FREQuency:FSTOp? 5-414
:SENSe{1-16}:FSEGMent{1-50}:FREQuency[:CW][:FIXed] <NRf> 5-415
:SENSe{1-16}:FSEGMent{1-50}:FREQuency[:CW][:FIXed]? 5-415
:SENSe{1-16}:FSEGMent{1-50}:MODBB:POWer:PORT{1-4} <NRf> 5-415
:SENSe{1-16}:FSEGMent{1-50}:MODBB:POWer:PORT{1-4}? 5-415
:SENSe{1-16}:FSEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude] <NRf> 5-416
:SENSe{1-16}:FSEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]? 5-416
:SENSe{1-16}:FSEGMent{1-50}:SPAntype <char> 5-416
:SENSe{1-16}:FSEGMent{1-50}:SPAntype? 5-416
:SENSe{1-16}:FSEGMent{1-50}:SWEep:POINt <NR1> 5-417
:SENSe{1-16}:FSEGMent{1-50}:SWEep:POINt? 5-417
:SENSe{1-16}:FSEGMent{1-50}[:STATe] <char> 5-417

:SENSe{1-16}:FSEGMent{1-50}[:STATe]? 5-417
:SENSe{1-16}:GRANging:AUTO[:STATe] <char> 5-418
:SENSe{1-16}:GRANging:AUTO[:STATe]? 5-418
:SENSe{1-16}:GRANging:MGSTate <char> 5-418
:SENSe{1-16}:GRANging:MGSTate? 5-418
:SENSe{1-16}:HOLD:FUNCtion <char> 5-419
:SENSe{1-16}:HOLD:FUNCtion? 5-419
:SENSe{1-16}:IMD:CONFigure 5-421
:SENSe{1-16}:IMD:CONFigure:DEFine[:STATe] <char> 5-422
:SENSe{1-16}:IMD:CONFigure:DEFine[:STATe]? 5-422
:SENSe{1-16}:IMD:CONFigure:STATus? 5-422
:SENSe{1-16}:IMD:DRVer:PORT? 5-422
:SENSe{1-16}:IMD:ENABled[:STATe] <char> 5-423
:SENSe{1-16}:IMD:ENABled[:STATe]? 5-423
:SENSe{1-16}:IMD:GRANging:AUTO[:STATe] <char> 5-423
:SENSe{1-16}:IMD:GRANging:AUTO[:STATe]? 5-423
:SENSe{1-16}:IMD:GRANging:MGSTate <char> 5-423
:SENSe{1-16}:IMD:GRANging:MGSTate? 5-423
:SENSe{1-16}:IMD:MODe <char> 5-424
:SENSe{1-16}:IMD:MODe? 5-424
:SENSe{1-16}:IMD:NCALibration:PERForm 5-424
:SENSe{1-16}:IMD:NCALibration[:STATe] <char> 5-424
:SENSe{1-16}:IMD:NCALibration[:STATe]? 5-424
:SENSe{1-16}:IMD:PARameter{1-16}:DISPlay <char> 5-425
:SENSe{1-16}:IMD:PARameter{1-16}:DISPlay? 5-425
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:ORDer <char> 5-425
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:ORDer? 5-425
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:PLANe <char> 5-426
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:PLANe? 5-426
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:TYPE <char> 5-426
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:REFerence:TYPE? 5-426
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:SIDE <char> 5-427
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:SIDE? 5-427
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:TYPE <char> 5-428
:SENSe{1-16}:IMD:PARameter{1-16}:RESPonse:TYPE? 5-428
:SENSe{1-16}:IMD:POINts <NRf> 5-428
:SENSe{1-16}:IMD:POINts? 5-428
:SENSe{1-16}:IMD:POWer:PORT:COUPle[:STATe] <char> 5-429
:SENSe{1-16}:IMD:POWer:PORT:COUPle[:STATe]? 5-429
:SENSe{1-16}:IMD:PRODuct:IFBW <NRf> 5-429
:SENSe{1-16}:IMD:PRODuct:IFBW? 5-429
:SENSe{1-16}:IMD:RECeiver:PORT:SELect <char> 5-429
:SENSe{1-16}:IMD:RECeiver:PORT:SELect? 5-429
:SENSe{1-16}:IMD:SPECtrum:DELTa <NRf> 5-430
:SENSe{1-16}:IMD:SPECtrum:DELTa? 5-430
:SENSe{1-16}:IMD:SPECtrum:FREQuency <NRf> 5-430
:SENSe{1-16}:IMD:SPECtrum:FREQuency? 5-430
:SENSe{1-16}:IMD:SPECtrum:ORDer <char> 5-430
:SENSe{1-16}:IMD:SPECtrum:ORDer? 5-430
:SENSe{1-16}:IMD:SPECtrum:SEGMented:SECond:ORDer[:STATe] <char> 5-431
:SENSe{1-16}:IMD:SPECtrum:SEGMented:SECond:ORDer[:STATe]? 5-431
:SENSe{1-16}:IMD:SPECtrum:TONE1:POWer <NRf> 5-431
:SENSe{1-16}:IMD:SPECtrum:TONE1:POWer? 5-431
:SENSe{1-16}:IMD:SPECtrum:TONE2:POWer <NRf> 5-431
:SENSe{1-16}:IMD:SPECtrum:TONE2:POWer? 5-431
:SENSe{1-16}:IMD:SPUR:AVOidance[:STATe] <char> 5-432
:SENSe{1-16}:IMD:SPUR:AVOidance[:STATe]? 5-432
:SENSe{1-16}:IMD:SWEPt:DELTa <NRf> 5-432
:SENSe{1-16}:IMD:SWEPt:DELTa? 5-432
:SENSe{1-16}:IMD:SWEPt:FREQuency:STARt <NRf> 5-432
:SENSe{1-16}:IMD:SWEPt:FREQuency:STARt? 5-432
:SENSe{1-16}:IMD:SWEPt:FREQuency:STOP <NRf> 5-433
:SENSe{1-16}:IMD:SWEPt:FREQuency:STOP? 5-433
:SENSe{1-16}:IMD:SWEPt:TONE1:POWer <NRf> 5-433
:SENSe{1-16}:IMD:SWEPt:TONE1:POWer? 5-433
:SENSe{1-16}:IMD:SWEPt:TONE2:POWer <NRf> 5-433
:SENSe{1-16}:IMD:SWEPt:TONE2:POWer? 5-433

:SENSe{1-16}:IMD:TONe:IFBW <NRf> 5-435
:SENSe{1-16}:IMD:TONe:IFBW? 5-435
:SENSe{1-16}:IMD:TONE1:SOURce <char> 5-434
:SENSe{1-16}:IMD:TONE1:SOURce? 5-434
:SENSe{1-16}:IMD:TONE2:SOURce <char> 5-435
:SENSe{1-16}:IMD:TONE2:SOURce? 5-435
:SENSe{1-16}:ISEGMent:ADD 5-436
:SENSe{1-16}:ISEGMent:AVERage:COUNt <NRf> 5-436
:SENSe{1-16}:ISEGMent:AVERage:COUNt? 5-436
:SENSe{1-16}:ISEGMent:BWIDth[:RESolution] <NRf> 5-437
:SENSe{1-16}:ISEGMent:BWIDth[:RESolution]? 5-437
:SENSe{1-16}:ISEGMent:CLEar 5-437
:SENSe{1-16}:ISEGMent:COUNt? 5-437
:SENSe{1-16}:ISEGMent:CWMODe[:STATe] <char> 5-438
:SENSe{1-16}:ISEGMent:CWMODe[:STATe]? 5-438
:SENSe{1-16}:ISEGMent:DATA? 5-438
:SENSe{1-16}:ISEGMent:DISPlay:AVERaging[:STATe] <char> 5-438
:SENSe{1-16}:ISEGMent:DISPlay:AVERaging[:STATe]? 5-438
:SENSe{1-16}:ISEGMent:DISPlay:IFBW[:STATe] <char> 5-439
:SENSe{1-16}:ISEGMent:DISPlay:IFBW[:STATe]? 5-439
:SENSe{1-16}:ISEGMent:DISPlay:POWer[:STATe] <char> 5-439
:SENSe{1-16}:ISEGMent:DISPlay:POWer[:STATe]? 5-439
:SENSe{1-16}:ISEGMent:FREQuency:FSTArt <NRf> 5-439
:SENSe{1-16}:ISEGMent:FREQuency:FSTArt? 5-439
:SENSe{1-16}:ISEGMent:FREQuency:FSTEp <NRf> 5-440
:SENSe{1-16}:ISEGMent:FREQuency:FSTEp? 5-440
:SENSe{1-16}:ISEGMent:FREQuency:FSTOp <NRf> 5-440
:SENSe{1-16}:ISEGMent:FREQuency:FSTOp? 5-440
:SENSe{1-16}:ISEGMent:FREQuency[:CW][:FIXed] <NRf> 5-441
:SENSe{1-16}:ISEGMent:FREQuency[:CW][:FIXed]? 5-441
:SENSe{1-16}:ISEGMent:INDex:ACTive:STARt? 5-441
:SENSe{1-16}:ISEGMent:INDex:ACTive:STOP? 5-441
:SENSe{1-16}:ISEGMent:INDex:STARt <NRf> 5-442
:SENSe{1-16}:ISEGMent:INDex:STARt? 5-442
:SENSe{1-16}:ISEGMent:INDex:STOP <NRf> 5-442
:SENSe{1-16}:ISEGMent:INDex:STOP? 5-442
:SENSe{1-16}:ISEGMent:MAXPoints? 5-443
:SENSe{1-16}:ISEGMent:MODBB:POWer:PORT{1-4} <NRf> 5-443
:SENSe{1-16}:ISEGMent:MODBB:POWer:PORT{1-4}? 5-443
:SENSe{1-16}:ISEGMent:POINt? 5-443
:SENSe{1-16}:ISEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]? 5-444
:SENSe{1-16}:ISEGMent:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]<NRf> 5-444
:SENSe{1-16}:ISEGMent:SPAntype <char> 5-444
:SENSe{1-16}:ISEGMent:SPAntype? 5-444
:SENSe{1-16}:ISEGMent:SWEep:MAXimize 5-445
:SENSe{1-16}:ISEGMent:SWEep:POINt <NR1> 5-445
:SENSe{1-16}:ISEGMent:SWEep:POINt? 5-445
:SENSe{1-16}:ISEGMent[:STATe] <char> 5-445
:SENSe{1-16}:ISEGMent[:STATe]? 5-445
:SENSe{1-16}:ISEGMent{1-50}:AVERage:COUNt <NRf> 5-446
:SENSe{1-16}:ISEGMent{1-50}:AVERage:COUNt? 5-446
:SENSe{1-16}:ISEGMent{1-50}:BWIDth[:RESolution] <NRf> 5-446
:SENSe{1-16}:ISEGMent{1-50}:BWIDth[:RESolution]? 5-446
:SENSe{1-16}:ISEGMent{1-50}:CWMODe[:STATe] <char> 5-447
:SENSe{1-16}:ISEGMent{1-50}:CWMODe[:STATe]? 5-447
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTArt <NRf> 5-447
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTArt? 5-447
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTEp <NRf> 5-447
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTEp? 5-447
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTOp <NRf> 5-448
:SENSe{1-16}:ISEGMent{1-50}:FREQuency:FSTOp? 5-448
:SENSe{1-16}:ISEGMent{1-50}:FREQuency[:CW][:FIXed] <NRf> 5-448
:SENSe{1-16}:ISEGMent{1-50}:FREQuency[:CW][:FIXed]? 5-448
:SENSe{1-16}:ISEGMent{1-50}:MODBB:POWer:PORT{1-4} <NRf> 5-449
:SENSe{1-16}:ISEGMent{1-50}:MODBB:POWer:PORT{1-4}? 5-449
:SENSe{1-16}:ISEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude] <NRf> 5-449
:SENSe{1-16}:ISEGMent{1-50}:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]? 5-449

:SENSe{1-16}:ISEGMent{1-50}:SPAntype <char> 5-450
:SENSe{1-16}:ISEGMent{1-50}:SPAntype? 5-450
:SENSe{1-16}:ISEGMent{1-50}:SWEep:POINt <NR1> 5-450
:SENSe{1-16}:ISEGMent{1-50}:SWEep:POINt? 5-450
:SENSe{1-16}:ISEGMent{1-50}[:STATe] <char> 5-451
:SENSe{1-16}:ISEGMent{1-50}[:STATe]? 5-451
:SENSe{1-16}:OFFSet:ADD 5-453
:SENSe{1-16}:OFFSet:BBModule:DELay:TIMe <NRf> 5-453
:SENSe{1-16}:OFFSet:BBModule:DELay:TIMe? 5-453
:SENSe{1-16}:OFFSet:BBModule:DELay[:STATe] <char> 5-454
:SENSe{1-16}:OFFSet:BBModule:DELay[:STATe]? 5-454
:SENSe{1-16}:OFFSet:BBModule:LEVeling:TYPe <char> 5-454
:SENSe{1-16}:OFFSet:BBModule:LEVeling:TYPe? 5-454
:SENSe{1-16}:OFFSet:BBModule:RCVR[:STATe] 5-454
:SENSe{1-16}:OFFSet:BBModule:RCVR[:STATe]? 5-454
:SENSe{1-16}:OFFSet:BBModule:SRC[:STATe] <char> 5-455
:SENSe{1-16}:OFFSet:BBModule:SRC[:STATe]? 5-455
:SENSe{1-16}:OFFSet:CLEar 5-455
:SENSe{1-16}:OFFSet:COMMon:OFFSet[:STATe] <char> 5-455
:SENSe{1-16}:OFFSet:COMMon:OFFSet[:STATe]? 5-455
:SENSe{1-16}:OFFSet:COUNt? 5-455
:SENSe{1-16}:OFFSet:DONe 5-456
:SENSe{1-16}:OFFSet:EXTernal{1-4}:CW[:STATe] <char> 5-457
:SENSe{1-16}:OFFSet:EXTernal{1-4}:CW[:STATe]? 5-457
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:DIVisor <NRf> 5-457
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:DIVisor? 5-457
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:MULTiplier <NRf> 5-457
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:MULTiplier? 5-457
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:OFFSet <NRf> 5-458
:SENSe{1-16}:OFFSet:EXTernal{1-4}[:FREQuency]:OFFSet? 5-458
:SENSe{1-16}:OFFSet:EXTModule:CONTrol:TYPe <char> 5-456
:SENSe{1-16}:OFFSet:EXTModule:CONTrol:TYPe? 5-456
:SENSe{1-16}:OFFSet:EXTModule[:STATe] <char> 5-458
:SENSe{1-16}:OFFSet:EXTModule[:STATe]? 5-458
:SENSe{1-16}:OFFSet:HIGHest:FREQuency? 5-458
:SENSe{1-16}:OFFSet:INTernal:PHASe <NRf> 5-460
:SENSe{1-16}:OFFSet:INTernal:PHASe? 5-460
:SENSe{1-16}:OFFSet:INTernal{1-2}:CW[:STATe] <char> 5-459
:SENSe{1-16}:OFFSet:INTernal{1-2}:CW[:STATe]? 5-459
:SENSe{1-16}:OFFSet:INTernal{1-2}:STATe 5-460
:SENSe{1-16}:OFFSet:INTernal{1-2}:STATe? 5-460
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:DIVisor <NRf> 5-459
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:DIVisor? 5-459
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:MULTiplier <NRf> 5-459
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:MULTiplier? 5-459
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:OFFSet <NRf> 5-460
:SENSe{1-16}:OFFSet:INTernal{1-2}[:FREQuency]:OFFSet? 5-460
:SENSe{1-16}:OFFSet:LOWest:FREQuency? 5-461
:SENSe{1-16}:OFFSet:MMModule:COMMon:MODe[:STATe] <char> 5-461
:SENSe{1-16}:OFFSet:MMModule:COMMon:MODe[:STATe]? 5-461
:SENSe{1-16}:OFFSet:MMModule:DELay:TIMe <char> 5-461
:SENSe{1-16}:OFFSet:MMModule:DELay:TIMe? 5-461
:SENSe{1-16}:OFFSet:MMModule:DELay[:STATe] <char> 5-462
:SENSe{1-16}:OFFSet:MMModule:DELay[:STATe]? 5-462
:SENSe{1-16}:OFFSet:MMModule:LEVeling:TYPe <char> 5-462
:SENSe{1-16}:OFFSet:MMModule:LEVeling:TYPe? 5-462
:SENSe{1-16}:OFFSet:MMModule:RCVR[:STATe] <char> 5-462
:SENSe{1-16}:OFFSet:MMModule:RCVR[:STATe]? 5-462
:SENSe{1-16}:OFFSet:MMModule:SRC[:STATe] <char> 5-463
:SENSe{1-16}:OFFSet:MMModule:SRC[:STATe]? 5-463
:SENSe{1-16}:OFFSet:MMModule:VCO[:FREQuency]:OVERrange <NRf> 5-463
:SENSe{1-16}:OFFSet:MMModule:VCO[:FREQuency]:OVERrange? 5-463
:SENSe{1-16}:OFFSet:PINVersion[:STATe] <char> 5-463
:SENSe{1-16}:OFFSet:PINVersion[:STATe]? 5-463
:SENSe{1-16}:OFFSet:RCVSource:CW[:STATe] <char> 5-464
:SENSe{1-16}:OFFSet:RCVSource:CW[:STATe]? 5-464
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:DIVisor <NRf> 5-464

:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:DIVisor? 5-464
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:MULTiplier <NRf> 5-464
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:MULTiplier? 5-464
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:OFFSet <NRf> 5-465
:SENSe{1-16}:OFFSet:RCVSource[:FREQuency]:OFFSet? 5-465
:SENSe{1-16}:OFFSet:RECEiver:CW[:STATe] <char> 5-465
:SENSe{1-16}:OFFSet:RECEiver:CW[:STATe]? 5-465
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:DIVisor <NRf> 5-465
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:DIVisor? 5-465
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:MULTiplier <NRf? 5-466
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:MULTiplier? 5-466
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:OFFSet <NRf> 5-466
:SENSe{1-16}:OFFSet:RECEiver[:FREQuency]:OFFSet? 5-466
:SENSe{1-16}:OFFSet:SPUR:AVOidance[:STATe] <char> 5-466
:SENSe{1-16}:OFFSet:SPUR:AVOidance[:STATe]? 5-466
:SENSe{1-16}:OFFSet:STARt <NRf> 5-467
:SENSe{1-16}:OFFSet:STARt? 5-467
:SENSe{1-16}:OFFSet:STOP <NRf> 5-467
:SENSe{1-16}:OFFSet:STOP? 5-467
:SENSe{1-16}:OFFSet[:STATe] <char> 5-468
:SENSe{1-16}:OFFSet[:STATe]? 5-468
:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay:TIMe <NRf> 5-468
:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay:TIMe? 5-468
:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay[:STATe] <char> 5-468
:SENSe{1-16}:OFFSet{1-50}:BBModule:DELay[:STATe]? 5-468
:SENSe{1-16}:OFFSet{1-50}:BBModule:LEVeling:TYPe <char> 5-469
:SENSe{1-16}:OFFSet{1-50}:BBModule:LEVeling:TYPe? 5-469
:SENSe{1-16}:OFFSet{1-50}:BBModule:RCVR[:STATe] <char> 5-469
:SENSe{1-16}:OFFSet{1-50}:BBModule:RCVR[:STATe]? 5-469
:SENSe{1-16}:OFFSet{1-50}:BBModule:SRC[:STATe] <char> 5-469
:SENSe{1-16}:OFFSet{1-50}:BBModule:SRC[:STATe]? 5-469
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}:CW[:STATe] <char> 5-470
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}:CW[:STATe]? 5-470
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:DIVisor <NRf> 5-470
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:DIVisor? 5-470
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:MULTiplier <NRf> 5-470
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:MULTiplier? 5-470
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:OFFSet <NRf> 5-471
:SENSe{1-16}:OFFSet{1-50}:EXTernal{1-4}[:FREQuency]:OFFSet? 5-471
:SENSe{1-16}:OFFSet{1-50}:EXTModule[:STATe] <char> 5-471
:SENSe{1-16}:OFFSet{1-50}:EXTModule[:STATe]? 5-471
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}:CW[:STATe] <char> 5-471
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}:CW[:STATe]? 5-471
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:DIVisor <NRf> 5-472
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:DIVisor? 5-472
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:MULTiplier <NRf> 5-472
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:MULTiplier? 5-472
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:OFFSet <NRf> 5-473
:SENSe{1-16}:OFFSet{1-50}:INTernal{1-2}[:FREQuency]:OFFSet? 5-473
:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe] <char> 5-473
:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe] <char> 5-476
:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe]? 5-473
:SENSe{1-16}:OFFSet{1-50}:MMModule:COMMon:MODe[:STATe]? 5-476
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe 5-477
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe <NRf> 5-474
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe? 5-474
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay:TIMe? 5-477
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe] 5-477
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe] <char> 5-474
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe]? 5-474
:SENSe{1-16}:OFFSet{1-50}:MMModule:DELay[:STATe]? 5-477
:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe 5-478
:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe <char> 5-475
:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe? 5-475
:SENSe{1-16}:OFFSet{1-50}:MMModule:LEVeling:TYPe? 5-478
:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe] 5-478
:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe] <char> 5-475

:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe]? 5-475
:SENSe{1-16}:OFFSet{1-50}:MMModule:RCVR[:STATe]? 5-478
:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe] 5-479
:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe] <char> 5-476
:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe]? 5-476
:SENSe{1-16}:OFFSet{1-50}:MMModule:SRC[:STATe]? 5-479
:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange 5-479
:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange <NRf> 5-476
:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange? 5-476
:SENSe{1-16}:OFFSet{1-50}:MMModule:VCO[:FREQuency]:OVERrange? 5-479
:SENSe{1-16}:OFFSet{1-50}:RCVSource:CW[:STATe] <char> 5-479
:SENSe{1-16}:OFFSet{1-50}:RCVSource:CW[:STATe]? 5-479
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:DIVisor <NRf> 5-480
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:DIVisor? 5-480
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:MULTiplier <NRf> 5-480
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:MULTiplier? 5-480
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:OFFSet <NRf> 5-481
:SENSe{1-16}:OFFSet{1-50}:RCVSource[:FREQuency]:OFFSet? 5-481
:SENSe{1-16}:OFFSet{1-50}:RECEiver:CW[:STATe] <char> 5-481
:SENSe{1-16}:OFFSet{1-50}:RECEiver:CW[:STATe]? 5-481
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:DIVisor <NRf> 5-482
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:DIVisor? 5-482
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:MULTiplier <NRf> 5-482
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:MULTiplier? 5-482
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:OFFSet <NRf> 5-483
:SENSe{1-16}:OFFSet{1-50}:RECEiver[:FREQuency]:OFFSet? 5-483
:SENSe{1-16}:OFFSet{1-50}:STARt <NRf> 5-483
:SENSe{1-16}:OFFSet{1-50}:STARt? 5-483
:SENSe{1-16}:OFFSet{1-50}:STOP <NRf> 5-484
:SENSe{1-16}:OFFSet{1-50}:STOP? 5-484
:SENSe{1-16}:RECeiver:CALibration:EXISt? <char> 5-485
:SENSe{1-16}:RECeiver:CALibration:STATe <char> 5-485
:SENSe{1-16}:RECeiver:CALibration:STATe? 5-485
:SENSe{1-16}:RECeiver:CALibration? <char> 5-485
:SENSe{1-16}:RECEiver:CONFiguration <char> 5-486
:SENSe{1-16}:RECEiver:CONFiguration? 5-486
:SENSe{1-16}:SEGMent:TYPe? 5-487
:SENSe{1-16}:SOURce{1-4}:EXTernal[:STATe] <char> 5-488
:SENSe{1-16}:SOURce{1-4}:EXTernal[:STATe]? 5-488
:SENSe{1-16}:SPUR:REDuction[:STATe] <char> 5-489
:SENSe{1-16}:SPUR:REDuction[:STATe]? 5-489
:SENSe{1-16}:SWEep:CW:POINt <NRf> 5-490
:SENSe{1-16}:SWEep:CW:POINt? 5-490
:SENSe{1-16}:SWEep:CW[:STATe] <char> 5-490
:SENSe{1-16}:SWEep:CW[:STATe]? 5-490
:SENSe{1-16}:SWEep:DELay <NRf> 5-491
:SENSe{1-16}:SWEep:DELay:TYPe <char> 5-491
:SENSe{1-16}:SWEep:DELay:TYPe? 5-491
:SENSe{1-16}:SWEep:DELay? 5-491
:SENSe{1-16}:SWEep:POINt <NRf> 5-491
:SENSe{1-16}:SWEep:POINt? 5-491
:SENSe{1-16}:SWEep:TIMe <NRf> 5-492
:SENSe{1-16}:SWEep:TIMe:DISPlay[:STATe] <char> 5-492
:SENSe{1-16}:SWEep:TIMe:DISPlay[:STATe]? 5-492
:SENSe{1-16}:SWEep:TIMe:STATe <char> 5-492
:SENSe{1-16}:SWEep:TIMe:STATe? 5-492
:SENSe{1-16}:SWEep:TIMe:TYPe <char> 5-493
:SENSe{1-16}:SWEep:TIMe:TYPe? 5-493
:SENSe{1-16}:SWEep:TIMe? 5-492
:SENSe{1-16}:SWEep:TYPe <char> 5-493
:SENSe{1-16}:SWEep:TYPe? 5-493
:SENSe{1-16}:TS3738:FREQuency:STARt <NRf> 5-494
:SENSe{1-16}:TS3738:FREQuency:STARt? 5-494
:SENSe{1-16}:TS3738:FREQuency:STOP <NRf> 5-494
:SENSe{1-16}:TS3738:FREQuency:STOP? 5-494
:SENSe{1-16}:TS3738:LO:DIVisor <NRf> 5-495
:SENSe{1-16}:TS3738:LO:DIVisor? 5-495

:SENSe{1-16}:TS3738:LO:MULTiplier <NRf> 5-495
:SENSe{1-16}:TS3738:LO:MULTiplier? 5-495
:SENSe{1-16}:TS3738:OML:WGType <char> 5-496
:SENSe{1-16}:TS3738:OML:WGType? 5-496
:SENSe{1-16}:TS3738:RECeiver <char> 5-497
:SENSe{1-16}:TS3738:RECeiver? 5-497
:SENSe{1-16}:TS3738:RF:DIVisor <NRf> 5-497
:SENSe{1-16}:TS3738:RF:DIVisor? 5-497
:SENSe{1-16}:TS3738:RF:MULTiplier <NRf> 5-498
:SENSe{1-16}:TS3738:RF:MULTiplier? 5-498
:SENSe{1-16}:TS3738:VDI:MODel <char> 5-498
:SENSe{1-16}:TS3738:VDI:MODel? 5-498
:SENSe{1-16}:TS3738:VDI:WGType <char> 5-499
:SENSe{1-16}:TS3738:VDI:WGType? 5-499
:SENSe{1-16}:TS3738[:STATe] <char> 5-500
:SENSe{1-16}:TS3738[:STATe]? 5-500
:SENSe{1-16}:TS3739:FREQuency:STARt 5-501
:SENSe{1-16}:TS3739:FREQuency:STARt? 5-501
:SENSe{1-16}:TS3739:FREQuency:STOP <NRf> 5-501
:SENSe{1-16}:TS3739:FREQuency:STOP? 5-501
:SENSe{1-16}:TS3739:LO:DIVisor <NRf> 5-502
:SENSe{1-16}:TS3739:LO:DIVisor? 5-502
:SENSe{1-16}:TS3739:LO:MULTiplier <NRf> 5-502
:SENSe{1-16}:TS3739:LO:MULTiplier? 5-502
:SENSe{1-16}:TS3739:OML:WGType 5-503
:SENSe{1-16}:TS3739:OML:WGType? 5-503
:SENSe{1-16}:TS3739:RECeiver 5-504
:SENSe{1-16}:TS3739:RECeiver? 5-504
:SENSe{1-16}:TS3739:RF:DIVisor <NRf> 5-504
:SENSe{1-16}:TS3739:RF:DIVisor? 5-504
:SENSe{1-16}:TS3739:RF:MULTiplier <NRf> 5-505
:SENSe{1-16}:TS3739:RF:MULTiplier? 5-505
:SENSe{1-16}:TS3739:VDI:MODel 5-505
:SENSe{1-16}:TS3739:VDI:MODel? 5-505
:SENSe{1-16}:TS3739:VDI:WGType <char> 5-506
:SENSe{1-16}:TS3739:VDI:WGType? 5-506
:SENSe{1-16}:TS3739[:STATe] <char> 5-507
:SENSe{1-16}:TS3739[:STATe]? 5-507
:SOURce:ALL:EXTernal:FTRIGger[:STATe] <char> 5-508
:SOURce:ALL:EXTernal:FTRIGger[:STATe]? 5-508
:SOURce{1-16}:EFFective:MODBB:POWer:PORT{1-4}? 5-509
:SOURce{1-16}:EFFective:POWer:PORT{1-4}[:LEVel][:IMMediate][:AMPlitude]? 5-509
:SOURce{1-16}:MODBB:POWer:PORT{1-4} <NRf> 5-510
:SOURce{1-16}:MODBB:POWer:PORT{1-4}? 5-510
:SOURce{1-16}:POWer:COMBiner[:STATe] <char> 5-511
:SOURce{1-16}:POWer:COMBiner[:STATe]? 5-511
:SOURce{1-16}:POWer:PORT:COUPle <char> 5-511
:SOURce{1-16}:POWer:PORT:COUPle? 5-511
:SOURce{1-16}:POWer:PORT{1-4}:ATTenuation <NRf> 5-515
:SOURce{1-16}:POWer:PORT{1-4}:ATTenuation? 5-515
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:EFFective:STOP? 5-515
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STARt? 5-515
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STARt<NRf> 5-515
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STEP? 5-515
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STOP <NRf> 5-516
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:POWer:STOP? 5-516
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:SINGle:POWer:EFFective:VALue? 5-516
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:SINGle:POWer:VALue <NRf> 5-516
:SOURce{1-16}:POWer:PORT{1-4}:LINear:MODBB:SINGle:POWer:VALue? 5-516
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:POINt <NRf> 5-517
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:POINt? 5-517
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STARt <NRf> 5-518
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STARt? 5-518
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STEP? 5-518
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STOP <NRf> 5-518
:SOURce{1-16}:POWer:PORT{1-4}:LINear:POWer:STOP? 5-518
:SOURce{1-16}:POWer:PORT{1-4}:LINear:REFerence:ATTenuation <NRf> 5-519

:SOURce{1-16}:POWer:PORT{1-4}:LINear:REFerence:ATTenuation? . . . 5-519
:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer:EFFective:VALue? . . . 5-519
:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer:VALue <NRf> . . . 5-519
:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer:VALue? . . . 5-519
:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer[:STATe] <char> . . . 5-520
:SOURce{1-16}:POWer:PORT{1-4}:LINear:SINGle:POWer[:STATe]? . . . 5-520
:SOURce{1-16}:POWer:PORT{1-4}:LINear:TEST:ATTenuation <NRf> . . . 5-520
:SOURce{1-16}:POWer:PORT{1-4}:LINear:TEST:ATTenuation? . . . 5-520
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed:FILe <string> . . . 5-528
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed:FILe? . . . 5-528
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed[:STATe] <char> . . . 5-528
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:DEEMbed[:STATe]? . . . 5-528
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed:FILe <string> . . . 5-529
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed:FILe? . . . 5-529
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed[:STATe] <char> . . . 5-529
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:FREQuency:EMBed[:STATe]? . . . 5-529
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed:FILe <string> . . . 5-530
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed:FILe? . . . 5-530
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed[:STATe] <char> . . . 5-529
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:DEEMbed[:STATe]? . . . 5-529
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed:FILe <string> . . . 5-530
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed:FILe? . . . 5-530
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed[:STATe] <char> . . . 5-530
:SOURce{1-16}:POWer:PORT{1-4}:NWExtract:POWer:EMBed[:STATe]? . . . 5-530
:SOURce{1-16}:POWer:PORT{1-4}:REFerence:ATTenuation <NRf> . . . 5-520
:SOURce{1-16}:POWer:PORT{1-4}:REFerence:ATTenuation? . . . 5-520
:SOURce{1-16}:POWer:PORT{1-4}:SLOPe <NRf> . . . 5-521
:SOURce{1-16}:POWer:PORT{1-4}:SLOPe? . . . 5-521
:SOURce{1-16}:POWer:PORT{1-4}:TEST:ATTenuation <NRf> . . . 5-521
:SOURce{1-16}:POWer:PORT{1-4}:TEST:ATTenuation? . . . 5-521
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:COLLect . . . 5-513
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:DATA <block> . . . 5-513
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:DATA? . . . 5-513
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:POWer:STARt <NRf> . . . 5-514
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:POWer:STARt? . . . 5-514
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection:POWer:STOP? . . . 5-514
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection[:STATe] <char> . . . 5-514
:SOURce{1-16}:POWer:PORT{1-6}:LINear:CORRection[:STATe]? . . . 5-514
:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:EFFective:STARt? . . . 5-516
:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:EFFective:STOP? . . . 5-517
:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:OFFSet <NRf> . . . 5-517
:SOURce{1-16}:POWer:PORT{1-6}:LINear:POWer:OFFSet? . . . 5-517
:SOURce{1-16}:POWer:PORT{1-7}:CORRection:COLLect . . . 5-511
:SOURce{1-16}:POWer:PORT{1-7}:CORRection:DATA <block> . . . 5-512
:SOURce{1-16}:POWer:PORT{1-7}:CORRection:DATA? . . . 5-512
:SOURce{1-16}:POWer:PORT{1-7}:CORRection:TARGet <NRf> . . . 5-512
:SOURce{1-16}:POWer:PORT{1-7}:CORRection:TARGet? . . . 5-512
:SOURce{1-16}:POWer:PORT{1-7}:CORRection[:STATe] <char> . . . 5-512
:SOURce{1-16}:POWer:PORT{1-7}:CORRection[:STATe]? . . . 5-512
:SOURce{1-16}:POWer:PORT{1-7}[:LEVel][:IMMediate][:AMPlitude] <NRf> . . . 5-521
:SOURce{1-16}:POWer:PORT{1-7}[:LEVel][:IMMediate][:AMPlitude]? . . . 5-521
:SOURce{1-16}:POWer:SLOPe[:STATe] <char> . . . 5-522
:SOURce{1-16}:POWer:SLOPe[:STATe]? . . . 5-522
:SOURce{1-16}:TS3739:ALCCal:FREQuency:POWer:TABLe{1-10}:ENTRy <NRf>, <NRf> . . . 5-523
:SOURce{1-16}:TS3739:ALCCal:FREQuency:POWer:TABLe{1-10}:ENTRy? . . . 5-523
:SOURce{1-16}:TS3739:ALCCal:LEVeltype <char> . . . 5-524
:SOURce{1-16}:TS3739:ALCCal:LEVeltype? . . . 5-524
:SOURce{1-16}:TS3739:ALCCal:POINt? . . . 5-524
:SOURce{1-16}:TS3739:ALCCal:PORT{1-4}:CORRection:COLLect? . . . 5-525
:SOURce{1-4}:EXTernal:ADDRess <NRf> . . . 5-526
:SOURce{1-4}:EXTernal:ADDRess? . . . 5-526
:SOURce{1-4}:EXTernal:FREQuency:MAXimum? . . . 5-526
:SOURce{1-4}:EXTernal:FREQuency:MINimum? . . . 5-527
:SOURce{1-4}:EXTernal:MODel? . . . 5-527
:SOURce{1-4}:EXTernal:POINt? . . . 5-527
:SOURce{1-4}:EXTernal:POWer <NRf> . . . 5-527
:SOURce{1-4}:EXTernal:POWer? . . . 5-527

:STATus:OPERation:CONDition? 5-531
:STATus:OPERation:ENABle <NRf> 5-531
:STATus:OPERation:ENABle? 5-531
:STATus:OPERation:NTRansition <NRf> 5-531
:STATus:OPERation:NTRansition? 5-531
:STATus:OPERation:PTRansition <NRf> 5-531
:STATus:OPERation:PTRansition? 5-531
:STATus:OPERation[:EVENt]? 5-532
:STATus:QUEStionable:CONDition? 5-533
:STATus:QUEStionable:ENABle <NRf> 5-533
:STATus:QUEStionable:ENABle? 5-533
:STATus:QUEStionable:LIMit:CONDition? 5-533
:STATus:QUEStionable:LIMit:ENABle <NRf> 5-533
:STATus:QUEStionable:LIMit:ENABle? 5-533
:STATus:QUEStionable:LIMit:NTRansition <NRf> 5-534
:STATus:QUEStionable:LIMit:NTRansition? 5-534
:STATus:QUEStionable:LIMit:PTRansition <NRf> 5-534
:STATus:QUEStionable:LIMit:PTRansition? 5-534
:STATus:QUEStionable:LIMit[:EVENt]? 5-534
:STATus:QUEStionable:NTRansition <NRf> 5-534
:STATus:QUEStionable:NTRansition? 5-534
:STATus:QUEStionable:PTRansition <NRf> 5-535
:STATus:QUEStionable:PTRansition? 5-535
:STATus:QUEStionable[:EVENt]? 5-535
:SYST:TIM:ELAP:INIT? – (System). 4-8
:SYST:TIM:ELAP:TURN? – (System) 4-8
:SYST:TIM:INIT? – (System) 4-8
:SYST:TIM:TURN? – (System). 4-8
:SYSTem:AUTocal:SWITch <char> 5-536
:SYSTem:AUTocal:SWITch? 5-536
:SYSTem:COMMunicate:GPIB:ADDRess <NRf> 5-537
:SYSTem:COMMunicate:GPIB:ADDRess? 5-537
:SYSTem:COMMunicate:GPIB:BBANd:PMETer:ADDRess 5-537
:SYSTem:COMMunicate:GPIB:BBANd:PMETer:ADDRess? 5-537
:SYSTem:COMMunicate:GPIB:DBANd:PMETer:ADDRess <NRf> 5-537
:SYSTem:COMMunicate:GPIB:DBANd:PMETer:ADDRess? 5-537
:SYSTem:COMMunicate:GPIB:EXTernal:TSET:ADDRess <NRf> 5-538
:SYSTem:COMMunicate:GPIB:EXTernal:TSET:ADDRess? 5-538
:SYSTem:COMMunicate:GPIB:FCOUNter:ADDRess <NRf> 5-538
:SYSTem:COMMunicate:GPIB:FCOUNter:ADDRess? 5-538
:SYSTem:COMMunicate:GPIB:PMETer:ADDRess <NRf> 5-538
:SYSTem:COMMunicate:GPIB:PMETer:ADDRess? 5-538
:SYSTem:COMMunicate:GPIB:PMETer:BBANd[:ADDRess] 5-539
:SYSTem:COMMunicate:GPIB:PMETer:BBANd[:ADDRess]? 5-539
:SYSTem:COMMunicate:GPIB:PMETer:DBANd[:ADDRess] 5-539
:SYSTem:COMMunicate:GPIB:PMETer:DBANd[:ADDRess]? 5-539
:SYSTem:COMMunicate:GPIB:PMETer:WBANd[:ADDRess] 5-539
:SYSTem:COMMunicate:GPIB:PMETer:WBANd[:ADDRess]? 5-539
:SYSTem:COMMunicate:GPIB:WBANd:PMETer:ADDRess <NRf> 5-540
:SYSTem:COMMunicate:GPIB:WBANd:PMETer:ADDRess? 5-540
:SYSTem:COMMunicate:TCPIP:ADDRess? 5-540
:SYSTem:COMMunicate:TCPIP:GATE? 5-540
:SYSTem:COMMunicate:TCPIP:HDW? 5-540
:SYSTem:COMMunicate:TCPIP:MASK? 5-540
:SYSTem:COMMunicate:TCPIP:PORT <NRf> 5-541
:SYSTem:COMMunicate:TCPIP:PORT? 5-541
:SYSTem:COMMunicate:USB:ADDRess? 5-541
:SYSTem:DUT:PROTection[:STATe] <char> 5-541
:SYSTem:DUT:PROTection[:STATe]? 5-541
:SYSTem:ERRor:CLEar 5-541
:SYSTem:ERRor:COUNt? 5-541
:SYSTem:ERRor:QUEue? 5-542
:SYSTem:ERRor[:NEXt]? 5-542
:SYSTem:EXTernal:TSET:FPORt:BIAS:TEES? 5-542
:SYSTem:EXTernal:TSET:FPORt:EXISt? 5-542
:SYSTem:EXTernal:TSET:FPORt:FREQuency:MAXimum? 5-542
:SYSTem:EXTernal:TSET:FPORt:FREQuency:MINimum? 5-543

:SYSTem:EXTernal:TSET:FPORt:MODel? 5-543
:SYSTem:EXTernal:TSET:FPORt:REVision? 5-543
:SYSTem:EXTernal:TSET:FPORt:VERSion? 5-543
:SYSTem:HOLD:BIAS[:STATe] <char> 5-543
:SYSTem:HOLD:BIAS[:STATe]? 5-543
:SYSTem:HOLD:POWer <NRf> 5-544
:SYSTem:HOLD:POWer? 5-544
:SYSTem:HOLD:RF[:STATe] <char> 5-544
:SYSTem:HOLD:RF[:STATe]? 5-544
:SYSTem:IFCalibration:AUTomatic[:STATe] <char> 5-544
:SYSTem:IFCalibration:AUTomatic[:STATe]? 5-544
:SYSTem:IFCalibration:DIGital:TRIGger 5-544
:SYSTem:IFCalibration:TRIGger 5-544
:SYSTem:POFF 5-545
:SYSTem:POINt:MAXimum <char> 5-545
:SYSTem:POINt:MAXimum? 5-545
:SYSTem:PORT:COUNt? 5-546
:SYSTem:PORT{1-4}:REFerence:ATTenuation <NRf> 5-546
:SYSTem:PORT{1-4}:REFerence:ATTenuation? 5-546
:SYSTem:PORT{1-4}:TEST:ATTenuation <NRf> 5-547
:SYSTem:PORT{1-4}:TEST:ATTenuation? 5-547
:SYSTem:POWerup:FILe <string> 5-547
:SYSTem:POWerup:FILe? 5-547
:SYSTem:POWerup:TYPe <char> 5-547
:SYSTem:POWerup:TYPe? 5-547
:SYSTem:PRESet 5-548
:SYSTem:PRESet:FILe <string> 5-548
:SYSTem:PRESet:FILe? 5-548
:SYSTem:PRESet:TYPe <char> 5-549
:SYSTem:PRESet:TYPe? 5-549
:SYSTem:PRESet:ZERo 5-550
:SYSTem:RETRace:RF[:STATe] <char> 5-551
:SYSTem:RETRace:RF[:STATe]? 5-551
:SYSTem:TIMe:ELAPsed:INITial? – (System) 4-8
:SYSTem:TIMe:ELAPsed:TURNon? – (System) 4-8
:SYSTem:TIMe:INITial? – (System) 4-8
:SYSTem:TIMe:TURNon? – (System) 4-8
:TRIGger[:SEQuence]:EXTernal:DELay <NRf> 5-552
:TRIGger[:SEQuence]:EXTernal:DELay? 5-552
:TRIGger[:SEQuence]:EXTernal:EDGe <char> 5-552
:TRIGger[:SEQuence]:EXTernal:EDGe? 5-552
:TRIGger[:SEQuence]:EXTernal:HANDshake[:STATe] <char> 5-553
:TRIGger[:SEQuence]:EXTernal:HANDshake[:STATe]? 5-553
:TRIGger[:SEQuence]:EXTernal:TYPe <char> 5-553
:TRIGger[:SEQuence]:EXTernal:TYPe? 5-553
:TRIGger[:SEQuence]:MANual:TYPe <char> 5-553
:TRIGger[:SEQuence]:MANual:TYPe? 5-553
:TRIGger[:SEQuence]:OUT[:STATe] <char> 5-553
:TRIGger[:SEQuence]:OUT[:STATe]? 5-553
:TRIGger[:SEQuence]:REMote:TYPe <char> 5-554
:TRIGger[:SEQuence]:REMote:TYPe? 5-554
:TRIGger[:SEQuence]:SEDTransfer[:STATe] <char> 5-554
:TRIGger[:SEQuence]:SEDTransfer[:STATe]? 5-554
:TRIGger[:SEQuence]:SOURce <char> 5-554
:TRIGger[:SEQuence]:SOURce? 5-554
:TRIGger[:SEQuence][:IMMediate][:REMote] 5-555
:TRIGger[:SEQuence][:REMote]:SINGle 5-557
*CLS 3-2
*DDT <Arbitrary Block> | <String> 3-2
*DDT? 3-2
*ESE <NRf> 3-3
*ESE? 3-3
*ESR? 3-3
*IDN? 3-3
*IST? 3-3
*OPC 3-4
*OPC? 3-6

*OPT? 3-6
*PRE <NRf> 3-6
*PRE? 3-6
*RST 3-7
*SRE <NRf> 3-8
*SRE? 3-8
*STB? 3-8
*TRG 3-9
*TST? 3-9
*WAI 3-10
ALCCALON <char> – (System) 4-3
ALCCALON? – (System). 4-3
FIND <string> – (System) 4-3
FRCVCALON <char> – (System). 4-3
FRCVCALON? – (System) 4-3
FRFCALON? – (System). 4-3
HELP <char> – (System) 4-3
IRCVCAL <arbitrary block> – (System) 4-4
IRCVCAL? <char> – (System) 4-4
ISDEBUG? – (System) 4-4
ISRCVCAL? – (System) 4-4
LANG <char> – (System) 4-5
LANG? – (System). 4-5
LIST – (System). 4-6
ONCS – (System) 4-6
ORCVCAL – (System). 4-6
RCVCAL <char1>, <char2> – (System) 4-6
RCVCALON <char1>, <char2> – (System) 4-6
RCVCALON? <char1> – (System) 4-6
RTL – (System) 4-6
TST? – (System) 4-7
TSTAIF? – (System) 4-7
TSTDECK? – (System) 4-7
TSTDSP? – (System). 4-7
TSTRES? – (System). 4-7
TSTSRC? – (System). 4-7

10410-00322 L

Anritsu Company
490 Jarvis Drive
Morgan Hill, CA 95037-2809
USA
http://www.anritsu.com