OmniBER XM network simulator

SCPI Remote Control Manual

Agilent Technologies

Agilent OmniBER XM

Network Simulator

SCPI Manual

Agilent Technologies

Notices

Manual Part Number

J7241-90002

Edition

Second edition, July 2003

Printed in UK

Agilent Technologies UK Limited
Data Networks Division
South Queensferry, West Lothian,
Scotland EH30 9TG

Warranty

The material contained in this document is provided "as is," and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as "Commercial computer software" as defined in DFAR 252.227-7014 (June 1995), or as a "commercial item" as defined in FAR 2.101(a) or as "Restricted computer software" as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause. Use, duplication or disclosure of Software is subject to Agilent Technologies' standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Safety Notices

CAUTION

A **CAUTION** notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a **CAUTION** notice until the indicated conditions are fully understood and met.

WARNING

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

AGILENT LICENSE TERMS

The following License Terms govern your use of the Licensed Materials unless you have a separate written agreement with Agilent, in which case, that written agreement will control and take precedence.

License Grant. Agilent grants you a non-exclusive license to use one copy of the Licensed Materials. With respect to the software portion of the Licensed Materials, "use" means to install, store, display, execute and use the software on the computer or device, or on the class or series of equipment, for which you have paid the corresponding license fee. If no fee is required, you may use the software on one computer or device. If the software is licensed for concurrent or network use, you may not allow more than the maximum number of authorized users to access and use the software concurrently.

License Restrictions. You may make copies or adaptations of the Licensed Materials only for archival purposes or only when copying or adaptation is an essential step in the authorized use of the Licensed Materials. You must reproduce all copyright notices in the original Licensed Materials on all permitted copies or adaptations. Except as provided here, you may not copy, modify or translate the Licensed Materials or disable any of its features. You may not copy the Licensed Materials onto any public or distributed network or service bureau. In addition, you may not lease, rent or sublicense the Licensed Materials without Agilent's prior written consent.

Upgrades. This license does not entitle you to receive upgrades, updates or technical support. Such services may be purchased separately. If the Licensed Materials include an upgrade to previously licensed material, your license in that material automatically terminates and you should destroy the previous content and any copies or adaptations.

Ownership. The Licensed Materials are owned and copyrighted by Agilent or its third party suppliers. Your license confers no title to, or ownership in, the Licensed Materials and is not a sale of any rights in the Licensed Materials. Agilent's third party suppliers may protect their rights in the event of any violation of these License Terms.

No Disassembly. You may not disassemble or decompile the Licensed Materials unless you obtain Agilent's prior written consent. In some jurisdictions, Agilent's consent may not be required for limited disassembly or decompilation. Upon request, you will provide Agilent with reasonably detailed information regarding any such permitted disassembly or decompilation.

High Risk Activities. The Licensed Materials are not specifically designed, manufactured or intended for use in the planning, construction, maintenance or direct operation of a nuclear facility, nor for use in on-line control or fail safe operation of aircraft navigation, control or communication systems, weapon systems or direct life support systems.

Transfer. You may transfer the license granted to you here provided that you deliver all the Licensed Materials to the transferee along with these License Terms. The transferee must accept these License Terms as a condition to any transfer. Your license to use the Licensed Materials will terminate upon transfer.

Termination. Agilent may terminate your license upon notice for failure to comply with any of these License Terms. Upon termination, you must immediately destroy the Licensed Materials, together with all copies, adaptations and merged portions in any form.

Export Requirements. The Licensed Materials may be subject to export control laws, including the U.S. Export Administration Regulations and other export laws and regulations of other countries. You may not export or re-export the Licensed Materials or any copy or adaptation in violation of any applicable laws or regulations. You certify that you are not on the U.S. Department of Commerce's Denied Persons List, the U.S. Department of Treasury's Specially Designated Nationals list or other government list prohibiting you from receiving the Licensed Materials.

U.S. Government Restricted Rights. Software and technical data rights granted to the federal government herein by these License Terms include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

METHODS OF REMOTE CONTROL....4

INTRODUCTION....5
CONTROLLING THE INSTRUMENT....6
TO INITIALIZE THE INSTRUMENT....7
REMOTE CONTROL HINTS & TIPS....8
SCPI COMMAND FORMAT....9
REMOTE CONTROL COMMANDS....13

USING SCPI WITH OMNIBER-XM....14

THE SESSION MANAGER....15
THE TEST SESSION....15
USING SCPI TO CONTROL AN OMNIBER XM SESSION....17

SMANAGER COMMANDS....21

SMANAGER:MODULE: COMMANDS....25

TSESSION COMMANDS....34

SYSTEM:MODULE COMMANDS....42

COMMON COMMAND REFERENCE....47

OUTPUT COMMANDS....48
SOURCE COMMANDS - TRANSMITTER COMMON COMMANDS....49
SOURCE COMMANDS - CLOCK FUNCTION COMMANDS....50
SOURCE COMMANDS - PATTERN COMMANDS....51
SOURCE COMMANDS – ERRORS & ALARM FUNCTIONS....53
ERROR INJECT....57
ERROR STRESS CONTROL....57
SENSE COMMANDS....59
SENSE COMMANDS – RECEIVER COMMON COMMANDS....60
SENSE COMMANDS - PATTERN COMMANDS....61
SENSE COMMANDS - TEST TIMING....62
SENSE COMMANDS – COMMON SENSE:DATA....66
RETURNED VALUES FOR SENSE:DATA....66
TOTAL TIME RESULTS....66
TOTAL RESULTS....67
SERVICE DISRUPTION RESULTS....67
COMMON SYSTEM COMMANDS....68
LOGGING - ERRORS....69
LOGGING - ALARMS....72
LOGGING - STATISTICS....75

IEEE COMMANDS 79

SDH COMMAND REFERENCE 82

SOURCE COMMANDS - TRANSMITTER SDH SETTINGS COMMANDS 83
SOURCE COMMANDS - TRANSMITTER SDH OVERHEAD SET UP 87
SOURCE COMMANDS - TRANSMITTER ERROR TEST FUNCTIONS...........................94
SOURCE COMMANDS - TRANSMITTER ALARM TEST FUNCTIONS.......................100
SOURCE COMMANDS - POINTER ADJUST TEST FUNCTIONS 103
SOURCE COMMANDS - FREQUENCY OFFSET TEST FUNCTIONS..........................104
SOURCE COMMANDS - APS MESSAGES 105
SOURCE COMMANDS - THRU MODE 106
SENSE COMMANDS - RECEIVER SDH SETTINGS..107
SENSE COMMANDS – SDH SENSE:DATA COMMANDS 114
RETURNED VALUES FOR SENSE:DATA...114
SDH ERROR COUNT RESULTS 114
SDH ERROR RATIO RESULTS..115
SDH POINTER ACTIVITY RESULTS..115
SDH POWER LEVELS 115
SDH ALARM SECONDS RESULTS 116
SDH ANALYSIS RESULTS...116
SENSE:DATA:TELECOM:GR253:ANALYSIS:ERDIP:MODE <BOOLEAN>, <PORTID>...117
SETS THE CURRENT ERDIP MODE FOR GR253. THE CORRESPONDING QUERY RETURNS THE CURRENT SETTING. 118
SENSE:DATA:TELECOM:GR253:ANALYSIS:ERDIP:MODE? <PORTID>.........118
FETCH SDH COMMANDS 119
SDH OVERHEAD BYTES 119
SDH LABELLED OVERHEAD BYTES..121
SDH OVERHEAD TRACE MESSAGES 122

SONET COMMAND REFERENCE 124

SOURCE COMMANDS - TRANSMITTER SONET SETTINGS COMMANDS 125
SOURCE COMMANDS - TRANSMITTER SONET OVERHEAD SET UP....................130
SOURCE COMMANDS - TRANSMITTER ERROR TEST FUNCTIONS.........................137
SOURCE COMMANDS - TRANSMITTER ALARM TEST FUNCTIONS........................143
SOURCE COMMANDS - POINTER ADJUST TEST FUNCTIONS 146
SOURCE COMMANDS - FREQUENCY OFFSET TEST FUNCTIONS..........................147
SOURCE COMMANDS - APS MESSAGES 148
SOURCE COMMANDS – THRU MODE 149
SENSE COMMANDS - RECEIVER SONET SETTINGS 150
SENSE COMMANDS – RESULT RETURNING COMMANDS 158
RETURNED VALUES FOR SENSE:DATA...158
SONET ERROR COUNT RESULTS 158
SONET ERROR RATIO RESULTS 159

SONet Pointer Activity Results 159
SONet Power Levels 159
SONet Alarm Seconds Results 160
FETCh SONet Commands 162
SONet Overhead Bytes 162
SONet Labelled Overhead Bytes 164
SONet Overhead Trace Messages 165

STATUS COMMANDS 166

Common Registers 166

STATUS REPORTING SPECIFIC TO SDH & SONET 168

Status Byte 171
Standard Event Status Register 172
QUEStionable Status Register 173
QUEStionable Status Register Commands 174
OPERation Status Register 175
OPERational Status Register Commands 176
INSTrument Status Register Commands 178
DATA Status Register Commands 180
SONet | SDH:PATH Status Register Commands 182
SONet | SDH:PATH:SUMMary Status Register Commands 184
SONet | SDH Port Status Register 185
SONet | SDH:PORT Status Register Commands 187
SONet | SDH Port Status Summary Register 188
SONet | SDH:PORT:SUMMary Status Register Commands 189

SCPI ERROR MESSAGES 190

No Error 190
Command Errors [-199, -100] 190
Execution Errors [-299, -200] 196
Query Errors [-399, -300] 204
Query Errors [-499, -400] 206
OmniBerXM SCPI Specific Errors [-2065, -2026] 208
OmniBerXM SCPI Specific Errors [-4000, -5000] 209

Methods of Remote Control

Introduction

The Instrument can be remotely controlled via a LAN connection. It is not possible to connect to the instrument using GPIB or RS-232-C.

LAN Provides a parallel interface that allows the connection of other instruments and controllers to the system for example: workstations; other instruments; other test sets that have a LAN interface.

Controlling the Instrument

The simplest way to verify the connection and become familiar with remote LAN operation is to send a few commands and observe their effect. In this example using a terminal we will:

- Start a new test session
- Connect to this new session
- Add a port to the new session
- Start the laser on this new port

Table 1-1 Example sequence of SCPI commands

Terminal Input	**Comment**
:SMAN:SESS:OPEN “OmniberXm”, online	Opens a new Session
:SMAN:SESS:LIST:OPEN?	Returns the new session label
:SMAN:SESS:SOCK? 1	Returns the socket number for the new session.
	Now create a new telnet connection with the new socket number.
:TSES:PORT:ADD <portid>	Adds the given port to the test session.
:SYST:ERR?	Checks for any error messages.
:OUTP:TEL:LAS ON, 010101	Starts the laser on the given port.

To Initialize the Instrument

Regardless of the current set up, the following command will initialize the Instrument. It sets the Instrument to the factory default settings and clears all registers. It will halt any applications currently running.

```
*RST
```

It is recommended that you do not rely on default settings but program each setting to what you require.

Remote Control Hints & Tips

This section gives some Hints & Tips on how to control the Instrument via remote control. Before writing any program to control the Instrument it may help to manually go through the steps required to set up the desired configuration. The order in which you should send the corresponding SCPI commands will usually follow the order in which you set up the Instrument from the Front Panel.

Default Settings

In general, default settings should not be assumed. It is recommended that each instrument setting should be explicitly programmed to the desired value.

Also commands that affect higher level settings such as Signal Rate or Payload Type should be sent before commands to set up lower level settings e.g. Pattern. This is because, in some cases, higher level setting changes can affect the values of lower settings.

Error Checking

It is recommended that, when sending SCPI commands to the Instrument, you also periodically send the SYSTem:ERRor? command to check for any Remote Control Errors reported by the instrument. This command returns 0, "No Error" if there are no errors in the error queue. If the SYSTem:ERRor? command is sent after every set up command then it makes debugging any reported errors much easier since you will know exactly which command caused the error.

The actual error queue within the Instrument can be cleared by sending the *CLS command.

Set up Delays

Even after the Instrument has accepted a SCPI command it may take some time for it to execute the requested operation (e.g. payload change, single error add). It is therefore recommended that at least 250 ms is allowed after the command has completed before expecting the instrument operation to occur.

Status Registers

Status registers in general are only updated every 100 ms by the Instrument. Therefore you should avoid reading them any faster than that since, although it does not do any harm if they are read faster, it means that less processing power will be available to update the display.

If you need to detect a status register bit that is momentarily changing state, avoid using the condition register in the hope of catching both states. Instead it is much better to program the PTRansition and NTRansition registers to catch the event of interest, and then use the event register to monitor for the event.

SCPI Command Format

Instrument functions such as making measurements, retrieving data, and querying status, are performed by stringing together SCPI "nodes" into commands. The SCPI commands are based on a hierarchical structure called a "subsystem" that comprises a top level "root" node and one or more lower-level nodes and their parameters as follows:

:INPut:TELecom:CODE <CMI> or <AMI> or <HDB3>
    :INPut is the root node
    :TELecom is a second level node
    :CODE is a third level node
    CMI, AMI and HDB3 are parameters of the third-level:CODE node.

Command Syntax

Commands are shown as a mix of upper and lowercase characters.

Commands can be abbreviated for shorter program line lengths. The uppercase characters define the abbreviated form of the command.

Commands are formed by linking the root node with lower-level nodes. A colon (:) is used to link nodes.
If the command requires a parameter, a space must separate the lowest level node and the parameter. If there is more than one parameter, a comma (,) is used to separate the parameters.

Examples of typical commands and their abbreviated form are shown below:

:INPut:TELecom:CODE AMI	full form
:INP:TEL:CODE AMI	abbreviated form
:SOURce:DATA:TELecom:PATTern PRBS23	full form
:SOUR:DATA:TEL:PATT PRBS23	abbreviated form

SCPI Long Form Command Headers

The general rule for SCPI long form command nodes that are greater than four characters in length is as follows:

Abbreviated short form mnemonics - the first four characters from the long form mode are used unless the fourth character is a vowel. In such cases, the vowel is dropped and only the first three characters are used.

If the node is four characters long then all four characters are used, irrespective of whether the fourth character is a vowel or not.

Linking Command Statements

Command statements can be linked using semicolons (;). For example:

:INPut:TELecom:CODE AMI;:SOURce:DATA:TELecom:PATTern PRBS23

Parameters

In this manual, parameters are shown in angled brackets < >. There are seven parameter types used in commands as listed in the table below.

Parameter Types	Description
<numeric>	All commonly used decimal numbers including optional signs, decimal points, and scientific notation. Examples are 123, 123E2, –123, –1.23E2, .123, .123E2 and1.2300E–01. Special cases include MINimum and MAXimum. A numeric parameter can also be specified in hex, octal, and/or binary. Examples are #H7B, #Q173 and #B11110111.
<boolean>	A single binary condition that is either true or false. Examples are ON, OFF, 1 and 0.

Parameter Types **Description**

<discrete> Values that are represented by a string of alphanumeric characters. Examples are INTernal and EXTernal.

<string> Any set of ASCII characters enclosed within single quotes or double quotes. Examples are '1111111111111111' and "0000000000000000".

<block> Used to transfer large quantities of related data. Blocks can be sent as **definite length blocks** (#<numeric><numeric>) or **indefinite length blocks** (#0).

<channel> A channel refers to a specific channel on an instrument port. A channel parameter is defined as a string which contains the ASCII representation of the STM1 start position e.g. “1”

<portid> A port refers to a specific port on an instrument. A port parameter is defined as a six digit code. The first two digits refer to the Chassis number, the second two refer to the module number and the final two refer to the port on the module. CCMMPP

<modulenumber> A Module number is the first for digits of a port number, CCMM. It refers to the module or blade upon which the ports are mounted. There are usually multiple port on any one module.

Occasionally, a command may use more than one parameter of the same type. When this occurs, the order of the parameters used in the command will map to the order of the individual parameter descriptions.

Remote Control Commands

The remote control commands in this manual have been grouped into Sections that relate to instrument functions. These have then been split into Subsystems

The Subsystems used in the Instrument are listed in the table below.

SCPI Subsystems

Instrument Functions	Subsystem
To control SIGNAL OUT port	:OUTPut
To control instrument coupling.	:INSTrument
To control the transmitter data.	:SOURce
To control the SIGNAL IN port.	:INPut
To control the receiver results and graphics	:SENSe
To obtain results directly accumulated by the instrument.	:FETCh
To control the instrument misc. functions eg data setting.	:SYSTem
To control Status Reporting.	:STATus
To control the instruments various sessions. **SMANager commands may only be used to communicate with the Session Manager and cannot be used within a session**. See the section on “Using SCPI to control an Omniber Xm Session” for more details.	:SMANager
To control a current Test Session within the instrument.	:TSESsion

Using SCPI With OmniBER-Xm

This chapter contains general information on SCPI commands, and detailed information on the commands that may be used to setup the OmniberXm.

The OmniberXm interface can be spilt into two main levels. The Session Manager oversees all the currently active sessions runing on the instrument and SCPI commands are provided specifically for this level to control sessions. Below this, individual Test Sessions have there own set of commands for controlling their paticular session. The following chapters examines these two layers in more detail.

The Session Manager

The main function of the Session Manager is to create and manage test sessions. The Session Manager is accessed via commands prefixed by **SMANager**. The Session Manager is the first entity which we connect to when controlling the Omniber Xm.

To connect to the Session Manager using the SCPI interface, a connection should be made via telnet port 9003. The session manager listens on this port for telnet connection and upon connection it will respond with the following

“Agilent Technologies, J724x Omniber-XM, GB999999999 SCPI telnet client on localhost”

Once a connection has been established to the Session Manager, we may then begin transmitting SMANager commmands in order to create, control or connect to existing test sessions.

The Test Session

A Test Session allows the user to interact with the test system by creating sessions. It provides information about the test session and allows the state information of a session to be reset, saved to a file or restored from a file.

Multiple test sessions can be created, each one controlling a different set of test modules. Each session has its own unique session ID and will run on its own individual socket.

In order to connect to a test session, we must use the session manager to determine which socket the current test is using. Once this has been obtained it is simply a matter of establishing another telnet client connection to the new socket ot port number. (the original IP address should remain the same).

Using SCPI to control an Omniber Xm Session

This section introduces the steps involved when using SCPI to control an instrument. This covers both creating a new session and connecting to an existing session. All SCPI connections to the instrument use a Telnet connection.

Running the Host Software

Before a session can be created the underlying host software or ‘Resource manager’ should be running.

Opening a SCPI Connection

- Go to DOS Command prompt and type "telnet localhost 9003"
- If you are trying to connect to the SCPI parser running on another machine then type "telnet <IP Address of other machine> 9003"
- If the connection is successful then will receive a "SCPI>" prompt.

Trouble-Shooting a failed SCPI connection

- If the connection failed, then open the Task manager (to do this click right mouse button on the task bar on the bottom of your windows screen and then select Task Manager from the options) and check if you find the entry for "Resourcemenager" under the processes tab.
- If not, run *Cleanup.exe* and try starting the *Resourcemanager* again.
- If that also doesn't work, it is possibe that either you are using an old version of *Resourcemanager* or your SCPI dll's are not registered correctly with the system.

- You may want to find how many *resourcemanager.exe* files you have on your system. It is possible that an old file may be lurking somewhere and this may be the one getting executed rather than the intended one (present in *C:\Program files\Agilent\Omniberxm\bin*)

- Also check your PATH settings (by typing PATH at the DOS prompt) to verify that *C:\Program files\Agilent\Omniberxm\bin* is defined in there.

Creating a New Test Session

To create a new test session on an OmniberXm

1. At the SCPI prompt in the telnet client window, type the following **SMANager:SESSion:OPEN “OmniberXm”, online** This will send a request to the Session Manager to create a new session.

2. Each time a SCPI command is entered, it is good practice to follow with a **SYSTem:ERRor?** Query to ensure that there we no faults with the previous command. Ensure that the open command executed successfully and move onto the next step.

3. Now that a new session has been created we must connect to it. In order to do this we must discover which socket the session is using and then connect to it via that socket. To determine the socket being used by our new session, type the following. **SMANager:SESSion:LIST:OPEN?** This will return a list of the currently open sessions in the form of numerics. The last number in this list will be the session label which you have just created.

4. To find the socket for this session, issue the following command. **SMANager:SESSion:SOCKet? n** Where n is the last number which was returned by the previous command. This socket query should then return two values. The first value is the connection for “String based” programs and should only be used by automated applications. The second number

refers to the standard SCPI connection which should be used for all manual SCPI control.

5. The final stage to this process is to now connect to the test session via the socket which has just been returned. To do this we use a similar approach to that employed in connecting to the resource manager. Go to Dos Command Prompt and type “**telnet localhost nnnn**” where nnnn is the socket number returned by the SCPI command above. A new Telnet window should then open which is the SCPI interface to the new Test Session. Your may then begin controlling the instrument via the SCPI commands.

Connecting to an Existing Session

To connect to an existing test session on an OmniberXm

1. In order to connect to an existing session you need to know which socket the session is using and then connect to it via that socket. At the SCPI prompt in the telnet client window, type the following **SMANager:SESSion:LIST:OPEN?** This will send a request to the Session Manager to list the currently open sessions. The command will return a list of the currently open session in the form of numerics.

2. Each time a SCPI command is entered, it is good practice to follow with a **SYSTem:ERRor?** Query to ensure that there we no faults with the previous command. Ensure that the open command executed successfully and move onto the next step.

3. To find the socket for this session, issue the following command. **SMANager:SESSion:SOCKet? n** Where n is the last number which was returned by the **LIST:OPEN** command. This socket query should then return two values. The first value is the connection for “String based” programs and should only be used by automated applications. The second number refers to the standard SCPI connection which should be used for all manual SCPI control.

4. The final stage to this process is to connect to the existing session via the socket which has just been returned. To do this you use a similar approach to that employed in connecting to the resource manager. Go to Dos Command Prompt and type "**telnet open localhost nnnn**" where nnnn is the socket number returned by the SCPI command above. A new Telnet window should then open which is the SCPI interface to the Existing Test Session. Your may then begin controlling the instrument via the SCPI commands.

5. If some ports have already been added to the session prior to our SCPI connection, then these must be specified explicitly. If you are aware of the ports currently added to the session the simply issue the following command for each port.

   TSESsion:PORT:ADD <portid>

If however you are unaware of the current ports being used by the session then you must perform the following steps.

1. Issues the command **TSESsion:PORT:LIST?** This will return a list of the port currently in the Session. The return types are in the form of port handles. This is a list of all the ports in the current session.

2. For each port we must then find out the attributes of that port. To do this we must issue the command **TSESsion:PORT:DETails? <portid>** This command will return the details for an individual portid. +101,+1

3. This return value may then be converted into a portid as follows. +101,+1 beocomes 010101.

4. This then becomes the parameter for **TSES:PORT:ADD <portid>** command. Repeat steps 2-4 for each port handle returned by the first command.

This is nessasery to sycronise the internal maps within the SCPI parser with the current set of ports.

SMANager Commands

The SMANager subsystem is used to control the operation of sessions within the instrument.

`SMANager:SESSion:OPEN <string> <discrete>`

<string>	<sessiontype>	Session Type (See API)
<discrete>	Online	Mode
	Offline	Mode

The command will open a Test Session. The mode controls whether the session is connected to actual modules (ONLINE) or just storing the configuration locally (OFFLINE). The corresponding query will return the Session ID of the Last test to be opened by this command.

`SMANager:SESSion:OPEN?`

Returns	<long>	Session Id

`SMANager:SESSion:CLOSE <numeric>`

<numeric>	<long>	Session Id

Closes the Test Session corresponding to the Session Id.

`SMANager:SESSion:LIST:OPEN?`

Returns	list<long>	list of Session Id

Returns the list of Session Id running at the moment in the system.

`SMANager:SESSion:LIST:TYPEs?`

Returns	list<string>	list of Session Id

Returns the list of the types of test sessions supported by your tester.

SMANager:SESSion:TYPE? <numeric>

<numeric>	<long>	Session Id

Takes as input, a session ID.

Returns the type of test session the given session ID is.

Returns	<string>	SessionType

.

SMANager:SESSion:CONText? <numeric>

<numeric>	<long>	Session Id

Takes as input, a session ID.

How the current test session was opened. Currently, the only context supported is: `AGT_SESSION_EXE`: The session is running as standalone, detached executable program.

Returns	<enum>	AGT_SESSION_EXE

SMANager:SESSion:SOCKet? <numeric>

<numeric>	<long>	Session Id

Takes as input, a session ID.

Returns the socket that the specified test session is currently on.

Returns	<long>	Socket Numer

SMANager:SESSion:LABel <numeric> <string>

<numeric>	<long>	Session Id
<string>	<Label>	Session Label

This command takes as input a Session ID and a Session Label with which the session ID should be associated.

SMANager:SESSion:LABel? <numeric>

<numeric>	<long>	Session Id

This command takes as input a Session ID and returns the associated session label.

<returns>	<string>	Session Label

SMANager:SESSion:PID? <numeric>

<numeric>	<long>	Session Id

This command takes as input a Session ID and returns the associated Process ID for it. If there are several users running test sessions and you need to kill a session through the Windows Task Manager, you can use this to identify a session's process ID.

<returns>	< long>	Process ID

SMANager:SESSion:GUI:CONNections? <numeric> <string>

<numeric>	<long>	Session Id
<string>	<session>	Session Type

This command takes as input a Session ID and a Session Type and returns a figure for the number of GUI connections of this type.

<returns>	< long>	Number Connected

`SMANager:SESSion:GUI:MCONnect <string> <numeric>`

<string>	<string>	Session Type
<numeric>	<long>	Maximum no. of connections

This command takes as input a Session TYPE and a value for the maximum number of connections. This value is then set within the instrument.

`SMANager:SESSion:GUI:MCONnect? <string>`

<numeric>	<string>	Session Type

This command takes as input a Session Type and return a value for the maximum number of connections for a session of that type.

<returns>	<long>	Maximum no. of connections

SMANager:MODule: Commands

The SMANager subsystem is used to control the operation of sessions within the instrument.

SMANager:MODule:UNLock <string>

<string>	<string>	The serial number of a test module.

Force unlock of a module Used to provide a mechanism so if for some reason the locks held by a particular test session are not cleaned up.

SMANager:MODule:REBoot:MODule <string>

<string>	<string>	The serial number of a test module.

Reboots a module. The module must be unlocked. Used to provide a mechanism to reboot a module for diagnostic purposes if it doesn't reboot automatically.

SMANager:MODule:REBoot:ALL

This command reboots all modules. All modules must be unlocked first.

SMANager:MODule:UPDate:ALL

This command is issued to instruct the ModuleManager to reallocate module numbers as a result of modules being added and removed. If any modules are locked or rebooting, the update will be deferred until the modules are either ready or marked as failed.

SMANager:MODule:UPDate:AUToupdate <boolean>

<boolean>	<boolean>	Enable / Disable

This command Enables/Disables automatic updating of module numbers. If disabled, the Module Manager will not automatically assign module numbers.

SMANager:MODule:UPDate:AUToupdate?

This command checks to see if automatic updating of module numbers enabled.

<returns>	<boolean>	Enable / Disable

SMANager:MODule:USModule <string>

<string>	<string>	The serial number of a test module.

The ‘Use Single Module’ command, instead of assigning module numbers to all modules, will assign the number 1 to a single module. If automatic updating is enabled, or any modules are locked, this request will fail. For manufacturing test.

SMANager:MODule:LIST?

Lists all numbered modules. Returns a list of numbers identifying the test modules that are currently connected and powered up. Modules are assigned incremental numbers, according to their order on the Event daisy chain.

<returns>	list<long>	A list of numbers identifying the test modules.

`SMANager:MODule:LALL?`

Return a list of all modules found in the test system. Returns a list of module serial numbers. A module's serial number is shown on the physical test module, on a sticker on the back panel. If you are simulating modules through Demo mode, the number is "`AGT_MODULE_x`" where x describes the type of module being simulated.

<returns>	list<string>	A list of serial numbers of test modules.

`SMANager:MODule:SERial? <modulenumber>`

<modulenumber>	CCBB	Module number

Return the serial number of a numbered module.

<returns>	<string>	Serial Number

`SMANager:MODule:NUMber? <string>`

<string>	<string>	Serial number

Return the module number of the module with the corresponding serial.

<returns>	<string>	Serial Number

`SMANager:MODule:LIST:SAVE`

The Save Module List command saves the current list of modules found in the system to the registry.

SMANager:MODule:LIST:SAVE?

The Get Saved Module List command returns the saved list of modules.

<returns>	List<string>	List of Serial Numbers

SMANager:MODule:LIST:NEW?

The List New Modules command returns a list of modules in the system which were not there the last time the list of modules was saved.

<returns>	List<string>	List of Serial Numbers

SMANager:MODule:LIST:MISSing?

The List Missing Modules command returns a list of modules which were in the system the last time the module was saved, which are not there now.

<returns>	list<string>	List of Serial Numbers

SMANager:MODule:DESCription? <string>

<string>	<string>	Serial Number

This command returns the type of the module and the number of ports available in that module.

<returns>	<discrete>	Module Type
	<long>	The number of test ports in the test module

SMANager:MODule:PTYPe? <string> <numeric>

<string>	<string>	Serial Number
<numeric>	<long>	Port Number

Returns the physical interface type for the port (indexed from 1 to N).

<returns>	<discrete>	Port Type

SMANager:MODule:NCPCount? <string>

<string>	<string>	Serial Number

Returns the number of NCPs in the module. Each NCP may control more than one port.

<returns>	<long>	The number of NCPs (Network Centric Processors)

SMANager:MODule:SYNChronised? <string>

<string>	<string>	Serial Number

Return TRUE if the module is synchronized with other modules.

<returns>	<boolean>	True / False

SMANager:MODule:DUMMy? <string>

<string>	<string>	Serial Number

Returns false if module is not a dummy module.

<returns>	<boolean>	True / False

SMANager:MODule:CHASsis:NUMBer? <string>

<string>	<string>	Serial Number

Returns the Chassis number of the chassis with the corresponding serial number.

<returns>	< long>	Chassis Number

SMANager:MODule:CHASsis:SLOT? <string>

<string>	<string>	Serial Number

Returns the Chassis Slot number the card is in of the chassis with the corresponding serial number.

<returns>	< long>	Chassis Slot Number

SMANager:MODule:ANNotation <string> <string>

<string>	<string>	Serial Number
<string>	<string>	Module Annotation

This command allows the user to set a string describing a module in the registry.

SMANager:MODule:ANNotation? <string>

<string>	<string>	Serial Number

This command allows the user to get a string describing a module in the registry.

<returns>	<string>	Module Annotation

SMANager:MODule:CMASter? <string>

<string>	<string>	Serial Number

This command allows the user to get a query wether the module being queried is the clock master.

<returns>	<boolean>	True / False

SMANager:MODule:IPADdress? <string> <numeric>

<string>	<string>	Serial Number
<numeric>	<long>	NCP Index

This command returns the IP address of the given NCP.

<returns>	<string>	IP Address

SMANager:MODule:IPADdress:PRIMary? <string>

<string>	<string>	Serial Number

This command returns the IP address of the primary NCP.

<returns>	<string>	IP Address

SMANager:MODule:IPADdress:HOST?

This command eturns the IP address of the primary NCP.

<returns>	<string>	IP Address

SMANager:MODule:STATe? <string>

<string>	<string>	Serial Number

This command returns the current state of the given module.

<returns>	<discrete>	AGT_MODULE_READY AGT_MODULE_LOCKED AGT_MODULE_REBOOTING

SMANager:MODule:LOCK? <string>

<string>	<string>	Serial Number

This command returns the session handle currently lock the module for a given serial number.

<returns>	<long>	A handle to the test session

SMANager:MODule:SDOWn? <string>

<string>	<string>	Serial Number

This command checks whether a module needs to be shut down before powering off.

<returns>	<boolean>	1- yes 0 - no

SMANager:MODule:SDOWn <string>

<string>	<string>	Serial Number

This command shuts down a module before powering it off. The module will enter the AGT_MODULE_SHUTTING_DOWN state; once it enters the AGT_MODULE_SHUTDOWN state, it's safe to power off.

SMANager:MODule:LED:FLASh <string>

<string>	<string>	Serial Number

This command flashes power LEDs to yellow at 2 Hz for 5 seconds for module with given serial number.

SMANager:MODule:LED:IP

This command shows the IP addresses of all modules on their LED displays.

TSESsion Commands

The TSESsion subsystem is used to control the current Test session within the instrument.

`TSESsion:OPEN <string> <discrete>`

<string>	<enum>	Session Type eg “OmniberXm”
<discrete>	ONLINE	Session Mode
	OFFLINE	Session Mode

This command is used to open a test session of the given type. The mode controls whether the session is connected to actual modules (ONLINE) or just storing the configuration locally (OFFLINE). The corresponding query will return the Session ID of the last test to be opened by this command.

`TSESsion:OPEN?`

<return>	<long>	Session Handle

`TSESsion:CLOSe`

This command is used to close the close this test session but fails if a GUI client is attached to it or a test is starting.

`TSESsion:FSCLose`

Force Session Close. CloseSession closes a test session but fails if a GUI client is attached to it or a test is starting. ForceCloseSession will close this test session, regardless of these conditions.

`TSESsion:NPORts?`

Number of Ports returns the number of ports used by this test session.

Returns	<long>	Number of Ports.

TSESsion:HANDle?

Returns a handle for this session. Returns zero if the session is not currently open.

Returns	<long>	Session Handle

TSESsion:TYPE?

Returns a type for this session.

Returns	<string>	Session Type

TSESsion:MODE?

Returns a mode for this session.

Returns	<discrete>	ONLINE OFFLINE

TSESsion:CONText?

Returns a context for this session.

Returns	<discrete>	AGT_SESSION_EXE AGT_SESSION_DLL

TSESsion:SAVE <string>

<string>	<filenname>	.xml file

This command is used to save all saveable state in the test session to the requested filename.Use the file extension .xml. By default, the file is saved in the directory " <rt_root>/RouterTester/config/<SessionType>" where <rt_root> is the directory into which RouterTester was installed, usually

C:/ProgramFiles/Agilent. To save in another location, specify the full directory path along with the drive designator (c:/ or d:/).

TSESsion:RESTore <string>

<string>	<filenname>	.xml file

This command is used to restore the state of the test session from the supplied file. For each interface whose persistent state exists in the given file, the session will reset its state to its default prior to restoring the state from file. The session will be restored on the original ports.

TSESsion:INTerface:SAVeable?

List Saveable Interfaces returns a list of the interfaces in the test session that can be saved.

Returns	list<string>	A list of API objects that may be saved.

TSESsion:RESet

This command will reset the test session to its default state.

TSESsion:RESet:INTerfaces <list of strings>

<string>	List <string>	A list of objects to reset

This command will reset a subset of the test session configuration The user lists the interfaces to reset. For each interface the session will reset its state to its default prior to restoring the state from file.

TSESsion:DISConnect

This command may be issued in order to force a disconnect from the current session.

The corresponding query returns whether this has already occurred.

TSESsion:DISConnect?

TSESsion:PORT: Commands

TSESsion:PORT:ADD <portid>

This command is used to add a test port to the current test session. Return a id for the port given by the port number. The port number correspond to the lettered ports on the units. For example, Port A is 1, port B is 2. So to add chassis 1, module 5, port B is "AddPort 010502". This portid can be used anywhere in the remainder of the test system to refer to the test port. When a port is added to the test session, it is automatically locked. To unlock a module, the ports must be removed from the test session. The operation will fail if the module is locked in another session. A port cannot be added to the system while a test is running.

TSESsion:PORT:ADD:LIST <list>

<list>	<portid>	A list separated by commas

This command is used to add a set of ports to the current test session. Adding ports as a set reduces the time penalty inccured by downloading the data to each port in the list simultanousely.

TSESsion:PORT:REMove <portid>

<numeric>	<long>	Port id

This command is used to remove a test port from the current test session. When a port is removed all state associated with the port will be lost. If the port is subsequently added again, all associated parameters will have returned to their defaults as if the port was part of a new session. A port cannot be removed from the system while a test is running.

TSESsion:PORT:REMove:LIST <list>

<string>	list<long>	List of Port handles

This command is used to remove a number of test ports from the current test session. Faster than removing each port individually, since the event lines must be re-segmented whenever a module is removed from the test session, and this can take a couple of seconds each time. The list should be comma separated. Eg 010101, 010201

TSESsion:PORT:LIST?

This command to list ports that are part of the current test session.

<returns>	<list>	List of Port ID's

TSESsion:PORT:DETails? <portid>

This command returns the module number and port number for the port with the corresponding portid.

<returns>	<long>	Module number
	<long>	Port number

TSESsion:PORT:DUMMy? <portid>

This command will return return true if the port is a dummy port (an instance of the embedded software running on the host PC).

<returns>	<boolean>	True / False

TSESsion:PORT:MODule:ADD <string>

<discrete>	<string>	Module Type

This command is used to add a new simulated module for use in demo mode. The module will be added to the end of the list. Can only be used in OFFLINE mode.

<returns>	list<long>	Port IDs

TSESsion:PORT:MODule:REMove <modulenumber>

<modulenumber>	CCBB	Module Number

This command is used to remove the simulated module at the location determined by ModulePosition where the first module is number 1 and the last module, number N. All subsequent modules will be removed as well (i.e., if module 3 is removed, module 3 to module N-1 are removed). Can only be used in OFFLINE mode.

TSESsion:PORT:MODule:PLISt? <modulenumber>

<modulenumber>	CCBB	Module Number

This command is used to list all the port handles that are currently contained within the given module.

TSESsion:PORT:COMMent <string> <portid>

<string>	<string>	Port Comment

This command is used to set information on how to configure the port. For example, "Connects to 192.10.3.2".

The Corresponding query is used to return the Port Comment.

`TSESsion:PORT:COMMent? <portid>`

<return>	<string>	Port Comment

SYSTem:MODule Commands

`SYSTem:MODule:LIST?`

This command returns the list of modules currently connected to the test system.

Returns	list<long>	A list of module numbers

`SYSTem:MODule:LAST?`

This command returns the number of the last module currently connected to the test system

Returns	<long>	module number

`SYSTem:MODule:DESCription <modulenumber>`

<modulenumber>	CCBB	Module Number

This command returns details of the module with the given module number. Returns the type of the module and the number of ports available in that module. Providing the port count enables clients to discover the number of ports available without having to hardcode details of specific module types.

Returns	<discrete>	The type of test port
	<long>	number of ports

`SYSTem:MODule:SYNChronised? <modulenumber>`

<modulenumber>	CCBB	Module Number

Returns true if the module must be synchronized with its neighbors.

Returns	<boolean>	True / False

SYSTem:CHASsis:NUMBer? <modulenumber>

<modulenumber>	CCBB	Module Number

For a chassis blade, this command returns its chassis number.

Returns	<long>	Chassis Number

SYSTem:CHASsis:SLOT? <modulenumber>

<modulenumber>	CCBB	Module Number

For a chassis blade, returns its slot number within the chassis.

Returns	<long>	Chassis Slot Number

SYSTem:TYPE? <portid>

Returns the type of the specified port on the module with the given module number. The ports on each module are numbered from 1 to N, where N is the port count returned by GetModuleDescription.

Returns	<discrete>	Port Type

SYSTem:MODule:TYPes? <modulenumber>

<modulenumber>	CCBB	Module Number

Returns a list of the module types supported by the module.

Returns	List<discrete>	A list of Module Types

SYSTem:MODule:TYPE <modulenumber> <discrete>

<modulenumber>	CCBB	Module number
<discrete>	<string>	Module type

Set the current module type. This method has no effect if the specified module type is the one currently exhibited by the module. The module type can only be set if the module is not currently owned by a test session.

SYSTem:MODule:LIMit?

If the number of modules supported by this session type is limited, this command returns the limit. If no limit is defined, return 0. Generally, the number of modules is not limited by a test session type, only by the hardware platform. Thus, most session types return a value of zero. An exception is PosAnalysis which supports only one test module.

<return>	<long>	module limit

SYSTem:MODule:SUPPorted? <modulenumber>

<modulenumber>	CCBB	Module Number

Returns true if the module is supported by this session type. Generally, most session types support most module types. Exceptions include PosAnalysis which supports only OC-192 modules, and OpticalTester which supports OC-48.

<return>	<boolean>	True / False

SYSTem:MODule:STATe? <modulenumber>

<modulenumber>	CCBB	Module Number

This command returns the current state of the given module.

<return>	<discrete>	AGT_MODULE_READY AGT_MODULE_LOCKED AGT_MODULE_REBOOTING AGT_MODULE_FAILED

SYSTem:MODule:LOCK? <modulenumber>

<modulenumber>	CCBB	Module Number

This command returns the session ID of the test session holding a lock on the given module number. Returns zero if there is no lock on the given module.

<return>	<long>	SessionID

SYSTem:MODule:LOCK:UPSTream? <numeric>

<numeric>	<long>	chassis number

This command returns the session ID of the test session which has locked this chassis for upstream synchronization. Returns zero if there is no lock.

<return>	<long>	SessionID

SYSTem:MODule:LOCK:DOWNstream? <numeric>

<numeric>	<long>	chassis number

This command returns the session ID of the test session which has locked this chassis for downstream synchronization. Returns zero if there is no lock.

<return>	<long>	SessionID

`SYSTem:MODule:LOCK:LIST?`

This command returns a list of the modules locked by this session.

<return>	list<long>	Module Numbers

`SYSTem:MODule:REQuired? <string>`

<string>	list<long>	list of modules

Given a list of modules to be selected, this command finds any additional modules which must also be selected to preserve synchronization across ports.

<return>	list<long>	list of modules

`SYSTem:MODule:UNAVailable? <string>`

<string>	list<long>	list of modules

Given a list of modules to be selected, this command finds any modules which are not available to be selected.

<return>	list<long>	list of modules

Common Command Reference

This chapter contains general and detailed information on the commands that are common for all types of instrument operations.

Output Commands

The OUTPut subsytem contains commands that control the characteristics of the instrument's output ports.

:OUTPut:TELecom:LASer <boolean> <portid>

<boolean> =	OFF	Select Laser Off
	ON	Select Laser On

Controls the state of the laser (ON or OFF) on the Optical module.

The corresponding query returns the state of the laser in discrete form as listed above.

:OUTPut:TELecom:LASer? <portid>

Returns:	<boolean>

:OUTPut:TELecom:LASer:ALL <boolean>

<boolean> =	OFF	Select All Lasers Off
	ON	Select All Lasers On

Controls the state of the lasers (ON or OFF) on all ports.

SOURce Commands - Transmitter Common Commands

:SOURce:DATA:TELecom:MODE <discrete> <portid>

<discrete> =	SDH	SDH Mode
	SONet	SONET Mode

Selects the transmitter mode.

The corresponding query returns the transmitter mode in discrete form listed above.

:SOURce:DATA:TELecom:MODE? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:THRU <boolean> <portid>

<boolean> =	0	Select Normal Mode
	1	Select Thru Mode

Selects/Deselects THRU mode.

The corresponding query returns the THRU mode state as a boolean value 0 or 1.

:SOURce:DATA:TELecom:THRU? <portid>

Returns:	<boolean>

SOURce Commands - Clock Function Commands

:SOURce:CLOCk:SOURce <discrete> <portid>

<discrete> =	INTernal	Internal
	RECovered	Recovered

Selects the Clock Source for the transmitter.

The corresponding query returns the transmitter clock source in discrete form as listed above.

:SOURce:CLOCk:SOURce? <portid>

Returns: <discrete>

SOURce Commands - Pattern Commands

:SOURce:DATA:TELecom:PATTern:TYPE <discrete> <channel> <portid>

<discrete> =	PRBS23	PRBS 2 ^{23} – 1 Pattern
	PRBS23inv	Inverted PRBS 23
	WORD	Word Pattern – User

Selects the transmitter payload Pattern Type, pseudo random sequence or a word type pattern.

The corresponding query returns the Pattern Type in discrete form as listed above.

:SOURce:DATA:TELecom:PATTern:TYPE? <channel> <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:PATTern:TYPE:WORD <discrete> <portid>

<discrete> =	USER	Selects user word

Selects the transmitter Payload Word Pattern type as either a preset word or a user generated word. This command is used if :SOURce:DATA:TELecom:PATTern:TYPE <discrete> is set to **Error! Reference source not found.**.

The corresponding query returns the Transmitter Payload Word type in discrete form as listed above.

:SOURce:DATA:TELecom:PATTern:TYPE:WORD? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:PATTern:TYPE:WORD:USER <numeric> <portid>

<numeric> =	32 bit value	Payload user word

Sets the Transmitter Payload User Word Pattern, Hex and binary patterns can be entered using the #h and #b forms, width of word is 32 bits.

This command is used when :SOURce:DATA:TELecom:PATTern:TYPE:WORD <discrete> is set to USER.

The corresponding query returns the Transmitter Payload User Word as an numeric.

:SOURce:DATA:TELecom:PATTern:TYPE:WORD:USER? <portid>

Returns:	<numeric>

SOURce Commands – Errors & Alarm Functions

Alarms

:SOURce:DATA:TELecom:ALARm <boolean> <portid>

<boolean> =	0 or OFF
	1 or ON

Enables and disables Alarm Generation.

The corresponding query returns the Alarm Generation state as 0 or 1.

:SOURce:DATA:TELecom:ALARm? <portid>

Returns:	<boolean>

:SOURce:DATA:TELecom:ALARm:ALL <boolean>

Enables and disables Alarm Generation on all ports.

Alarm Stress Control

:SOURce:DATA:TELecom:ALARm:MODE <discrete> <portid>

<discrete> =	MANual	Manual Alarm
	PULSed	Pulsed Alarm
	TIMed	Timed Alarm

Before any alarms may be transmited, the mode for the alarm must be set on the given port. The alarm mode can be either a Manual setting, Pulsed or Timed.

The corresponding query returns the alarm mode for a given port.

SOURce:DATA:TELecom:ALARm:MODE? <portid>

Timed Alarm Periods

:SOURce:DATA:TELecom:ALARm:STRess:TIMed <numeric> <numeric> <numeric> <portid>

<numeric> =	1 – 10 000	On Period
<numeric> =	1 – 10 000	Off Period
<numeric> =	1 – 10 000	Repeat Count

Used to setup the parameters for a timed alarm burst. Off Period may be 0 if RepeatCount = 1 to give a single timed burst of the alarm.

The Default parameters will be a single burst of 1 second (OnPeriod = 1, OffPeriod = 0, RepeatCount = 1).

The corresponding query returns the On, Off Period and Repeat Count.

:SOURce:DATA:TELecom:ALARm:STRess:TIMed? <portid>

<returns> =	<numeric>	On Period

	<numeric>	Off Period
	<numeric>	Repeat Count

:SOURce:DATA:TELecom:ALARm:STRess:TIMed:STARt <portid>

This command triggers the alarm burst using the setting specified in the above command.

:SOURce:DATA:TELecom:ALARm:STRess:TIMed:STOP <portid>

This command causes the alarm burst to cease.

The corresponding query returns a boolean indicating wether the alarm burst is running or not.

:SOURce:DATA:TELecom:ALARm:STRess:TIMed:RUNNing? <portid>

<returns> =	<boolean>	On, Off

:SOURce:DATA:TELecom:ALARm:STRess:PULSe <numeric> <portid>

<numeric> =	1-64	Pulse Count

Defines the pulse count for a given port. The default value for Pulse Count is 1.

The corresponding query returns pulse count from a given port.

:SOURce:DATA:TELecom:ALARm:STRess:PULSe? <portid>

<returns> =	<numeric>	Pulse Count

```
:SOURce:DATA:TELecom:ALARm:STRess:PULSe:STARt <portid>
```

This command starts the pulse burst. The pulsed burst is active for such a short time that there is no requirement for a stop or status command.

Error Inject

:SOURce:DATA:TELecom:ERRor:SINGle <portid>

Injects a single error on a given port.

There is no corresponding query.

:SOURce:DATA:TELecom:ERRor:SINGle:ALL

Injects a single error on all ports.

There is no corresponding query.

Error Stress Control

:SOURce:DATA:TELecom:ERRor:MODE <discrete> <portid>

<discrete> =	`MANual`	`Manual Error`
	`TIMed`	`Timed Error`

Before any errors may be transmited, the mode for the error must be set on the given port. The error mode can be either a Manual setting or Timed.

The corresponding query returns the error mode for a given port.

SOURce:DATA:TELecom:ERRor:MODE? <portid>

:SOURce:DATA:TELecom:ERRor:STRess:TIMed <numeric> <numeric> <numeric> <portid>

<numeric> =	1 – 10 000	On Period
<numeric> =	1 – 10 000	Off Period
<numeric> =	1 – 10 000	Repeat Count

Used to setup the parameters for a timed error burst. Off Period may be 0 if RepeatCount = 1 to give a single timed burst of the error.

The Default parameters will be a single burst of 1 second (OnPeriod = 1, OffPeriod = 0, RepeatCount = 1).

The corresponding query returns the On, Off Period and Repeat Count.

:SOURce:DATA:TELecom:ERRor:STRess:TIMed? <portid>

<returns> =	<numeric>	On Period
	<numeric>	Off Period
	<numeric>	Repeat Count

:SOURce:DATA:TELecom:ERRor:STRess:TIMed:STARt <portid>

This command triggers the error burst using the setting specified in the above command.

:SOURce:DATA:TELecom:ERRor:STRess:TIMed:STOP <portid>

This command causes the error burst to cease.

The corresponding query returns a boolean indicating wether the error burst is running or not.

:SOURce:DATA:TELecom:ERRor:STRess:TIMed:RUNNing? <portid>

<returns> =	<boolean>	On, Off

SENse Commands

This chapter contains detailed information on the commands that are common to both SDH & SONET for the SENSE subsystem.

SENSe Commands – Receiver Common Commands

:SENSe:DATA:TELecom:MODE <discrete> <portid>

<discrete> =	SDH	SDH Mode
	SONet	SONET Mode

Selects the receiver mode.

The corresponding query returns the receiver mode in discrete form listed above.

:SENSe:DATA:TELecom:MODE? <portid>

Returns:	<discrete>

SENSe Commands - Pattern Commands

:SENSe:DATA:TELecom:PATTern:TYPE <discrete> <channel> <portid>

<discrete> =	PRBS23	PRBS $2^{23} – 1$ Pattern
	PRBS23inv	Inverted PRBS 23
	WORD	WORD - User Pattern
	LIVE	Live Traffic

Selects the receiver payload pattern type.

The corresponding query returns the receiver payload data pattern type in discrete form as listed above

:SENSe:DATA:TELecom:PATTern:TYPE? <channel> <portid>

Returns: <discrete>

:SENSe:DATA:TELecom:PATTern:TYPE:WORD <discrete> <portid>

<discrete> =	USER	Selects user word

Selects the Expected Receiver Payload Word Pattern type a user generated word.

The corresponding query returns the Transmitter Payload Word type in discrete form as listed above.

:SENSe:DATA:TELecom:PATTern:TYPE:WORD? <portid>

Returns: <discrete>

:SENSe:DATA:TELecom:PATTern:TYPE:WORD:USER <numeric> <portid>

<numeric> =	32 bit	User Word Pattern

Sets the Receiver Payload User Word Pattern. The prefix of #b or #h can be used to specify binary or hex respectively, width of the user word is 32 bits.

The corresponding query returns the Receiver Payload User Word as an integer value.

:SENSe:DATA:TELecom:PATTern:TYPE:WORD:USER? <portid>

Returns:	<numeric>

SENSe Commands - Test Timing

:SENSe:DATA:TELecom:TEST <boolean>

<boolean> =	0 or OFF	Stop the current test period
	1 or ON	Start a new test period

Start/Stop the test

The corresponding query returns the test state as a discrete value. This may be one of four values; STOPPED: The test is idle. STARTING: All ports are preparing to start traffic generation and statistics collection synchronously. RUNNING: Traffic generation and statistics collection is in progress. STOPPING: Traffic generation has stopped, but residual statistics collection is continuing for the duration of the trickle time.

If the GUI is active on the current session then the function cannot be used if the laser is turned off on any of the ports.

:SENSe:DATA:TELecom:TEST?

Returns:	<discrete>	STOPPED
		STARTING

		RUNNING
		STOPPING

SENSe:DATA:TELecom:TEST:PORT:SELect <list>

<list> =	Portid	A list of Port ID's

This command is used to select the ports to from which statistics will be collected during the test period. Repeating the command with different port parameters will result in the previously selected ports being replaced. The corresponding query returns the currently selcted test ports.

SENSe:DATA:TELecom:TEST:PORT:SELect?

<return> =	<list>	A list of Port ID's

SENSe:DATA:TELecom:TEST:PORT:ANALysis:MODE <portid> <discrete>

<discrete> =	G828 GR253 NONE	Selects the Analysis Mode.

Selects the Analysis type for a given port. The port must first be added via the **SENSe:DATA:TELecom:TEST:PORTs <list>** command.

The corresponding query returns the chosen analysis type for a given port.

SENSe:DATA:TELecom:TEST:PORT:ANALysis:MODE? <portid>

Returns:	<discrete>

:SENSe:DATA:TELecom:TEST:TYPE <discrete>

<discrete> =	MANual	Manual Test period
	SINGle	SINGle Single Test period

Selects the type of test period. If SING is selected, the duration is set using :SENSe:DATA:TELecom:TEST:PERiod. The corresponding query returns the test type in discrete form, as listed above.

:SENSe:DATA:TELecom:TEST:TYPE?

Returns: <discrete>

:SENSe:DATA:TELecom:TEST:PERiod <numeric>, <numeric>, <numeric>, <numeric>

<numeric> =	0 to 6	Day
<numeric> =	0 to 23	Hour
<numeric> =	0 to 59	Minute
<numeric> =	0 to 59	Second

Sets the duration of the test period. Is only valid when :SENSe:DATA:TELecom:TEST:TYPE is set to SINGle.The corresponding query returns the test duration as described above.

:SENSe:DATA:TELecom:TEST:PERiod?

Returns: <numeric>, <numeric>, <numeric>, <numeric>

:SENSe:DATA:TELecom:TEST:STARt?

This command returns the time at which the command was started in the form of Year, Month, Day, Hour, Minute, Second.

Returns: <numeric>, <numeric>, <numeric>, <numeric>, <numeric>,<numeric>

:SENSe:DATA:TELecom:TEST:ELAPsed?

This command returns the time which has elapsed since the test was started. The command will return of Hours, Minutes, Seconds.

Returns: <numeric>, <numeric>, <numeric>

:SENSe:DATA:TELecom:SDGT:TIME <numeric> <portid>

<numeric> = LONG 100ms – 1600ms

This command sets the length of the Service Disruption Guard Time. It can be a value in the range 100ms to 1600ms in 1ms steps. This is accurate to +0.5ms.

The corresponding query returns the Service Disruption Guard Time.

:SENSe:DATA:TELecom:SDGT:TIME? <portid>

Returns: <numeric>

:SENSe:DATA:TELecom:SDGT:DEFault <portid>

Set the Service Disruption Guard Time to the default. (200ms)

SENSe Commands – Common SENSe:DATA

Common Results

Each of the SENSe:DATA commands may be applied to a given set of ports. To select the set of ports to monitor tests on, the following command should be used,

SENSe:DATA:TELecom:TEST:PORT:SELect <list>

In order to select a certain channel or group of channels, then the following command should be used.

:SENSe:DATA:TELecom:SONet:CHANnels:EQUipped <string> <portid>

See above commands for more details.

Returned values for SENSe:DATA

Each of the SENSe:DATA commands returns a set of floating point values which represent the values for each of the channels selected. The data returned is in the form of a comma separated list.

Total Time Results

:SENSe:DATA? <string> <portid>

<string> =

“ETIMe”	Returns elapsed time
“ASEConds:LOS”	Loss of Signal
“ASEConds:PSL”	Pattern Sync Loss

Total Results

:SENSe:DATA? <string> <portid>

<string> =

“ECOunt:BIT”	Bit Error Count
“ERATio:BIT”	Bit Error Ratio

Service Disruption Results

:SENSe:DATA? <string> <portid>

<string> =

“SDTest:COUNt:MAX”	Largest disruption in last period
"SDTest:COUNt:LAST"	Length of last disruption
"SDTest:COUNt:TOTal"	Total number of disruptions

Common SYSTem Commands

:SYSTem:PRESet

This command resets the instrument state. It performs the same task as *RST.

:SYSTem:SERial?

Returns:	<string>

The serial number is returned as a string in the form "GBnnnnnnnn". GB signifies the country of origin (Great Britain).

:SYSTem:VERSion?

Returns:	<string> =	YYYY.V

Returns the revision state of the SCPI remote control.

The revision state is returned in the form YYYY.V. YYYY signifies the year and V signifies the revision number.

:SYSTem:ERRor?

Returns:	<numeric>,<string>

Requests the Instrument remote control Error status.

The error status is returned as a numeric value and a string containing a description of the error.

Logging - Errors

Records selected erros to a text file in real time, as they are being measured.

SYSTem:LOGGing:ERRor <boolean>

<boolean>	ON, OFF	logging is enabled or disabled

Enables or disables logging to a text file. The corresponding query returns the current logging status.

SYSTem:LOGGing:ERRor?

Returns:	<boolean>

SYSTem:LOGGing:ERRor:FILE <string>

<string>	File name

Sets the file in which the loggine data will be stored. The tester creates the file when the test starts. The default directory is

"c:/Program Files/Agilent/RouterTester/data/<SessionType>"

where <SessionType> is type of session (for example, IpPerformance).

If you want to save files to another location, you must specify the full directory path including the drive designator (c:/). Use forward slashes (/), not backslashes (\).

Caution: If you save to an existing log file, the new data replaces the old data.

Statistics are saved in CSV (Comma Separated Values) format. Files should (but are not required to) have this .csv extension. By using this extension, Windows applications that can open or import these files (for example, Microsoft Excel) will be able to recognize them.

The corresponding query returns the current file name.

`SYSTem:LOGGing:ERRor:FILE? <string>`

Returns: <string>

`SYSTem:LOGGing:ERRor:PORTs <string>`

<string>	PortID, PortID	A List of Port ID’s

Takes in a list of port ID’s for the test ports for which you whish to log statistics.

The corresponding query return a list of selected ports.

SYSTem:LOGGing:ERRor:PORTs?

Returns: <string>

SYSTem:LOGGing:ERRor:AVAilable?

Returns a list of the currently available Error types which may be selected for logging.

SYSTem:LOGGing:ERRor:SELect <string>

<string>	CV-S(B1)	Sonet B1 Error Count
	CV-L(B2)	Sonet B2 Error Count
	REI-L	REIL Error Count
	SEF	SEF Errored Second
	CV-P(B3)	B3 Error Count
	BIT	Bit Error Count
	REI-P	REI P Error Count
	SrvDisCount	Service Disruption Count
	SrvDisLastTime	Service Disruption Last Time
	SrvDisMaxTime	Service Disruption Max Time
	B1	SDH B1
	B2	SDH B2
	MS-REI	MS-REI Error Count
	OOF	OOF Errored Second
	B3	SDH B3 Error Count
	HP-REI	HP-REI Error Count

This command accepts a list of errors as a comma separated list which defines which errors are to be logged.

The corresponding query returns a common separated list defining which errors have been selected for logging.

SYSTem:LOGGing:ERRor:SELect?

Returns: <string>

Logging - Alarms

Records selected alarms to a text file in real time, as they are being measured.

SYSTem:LOGGing:ALARm <boolean>

<boolean>	ON, OFF	logging is enabled or disabled

Enables or disables logging to a text file. The corresponding query returns the current logging status.

SYSTem:LOGGing:ALARm?

Returns:	<boolean>

SYSTem:LOGGing:ALARm:FILE <string>

<string>	File name

Sets the file in which the loggine data will be stored. The tester creates the file when the test starts. The default directory is

"c:/Program Files/Agilent/RouterTester/data/<SessionType>"

where <SessionType> is type of session (for example, IpPerformance).

If you want to save files to another location, you must specify the full directory path including the drive designator (c:/). Use forward slashes (/), not backslashes (\).

Caution: If you save to an existing log file, the new data replaces the old data.

Statistics are saved in CSV (Comma Separated Values) format. Files should (but are not required to) have this .csv extension. By using this extension, Windows applications that can open or import these files (for example, Microsoft Excel) will be able to recognize them.

The corresponding query returns the current file name.

SYSTem:LOGGing:ALARm:FILE?

Returns: <string>

SYSTem:LOGGing:ALARm:PORTs <string>

<string>	PortID, PortID	A List of Port ID’s

Takes in a list of port ID’s for the test ports for which you wish to log statistics.

The corresponding query return a list of selected ports.

SYSTem:LOGGing:ALARm:PORTs?

Returns: <string>

SYSTem:LOGGing:ALARm:AVAilable?

Returns a list of the currently available alarm types which may be selected for logging.

SYSTem:LOGGing:ALARm:SELect <string>

<string>	LOS	LOS
	LOF	LOF
	AIS-L	Sonet AISL
	RDI-L	Sonet RDIL

UNEQ-P	Path Unequipped
LOP-P	LOP Alarm
PDI-P	PDI-p Alarm
AIS-P	AIS-P Alarm
RDI-P	RDI-P Alarm
PSL	PSL Alarm
MS-AIS	MS-AIS Alarm
MS-RDI	MS-RDI Alarm
HP-UNEQ	HP-Unequipped
AU-LOP	AU-LOP
HP-PDI	HP-PDI
AU-AIS	AU-AIS
HP-RDI	HP-RDI
PSL	PSL
PTR-ACT	Ptr Activity Errored Second

This command accepts a list of alarms as a comma separated list which defines which alarms are to be logged.

The corresponding query returns a common separated list defining which alarms have been selected for logging.

SYSTem:LOGGing:ALARm:SELect?

Returns: <string>

Logging - STATistics

Records selected statistics to a text file in real time, as they are being measured.

SYSTem:LOGGing:STATistics <boolean>

<boolean>	ON, OFF	logging is enabled or disabled

Enables or disables logging to a text file. The corresponding query returns the current logging status.

SYSTem:LOGGing:STATistics?

Returns: <boolean>

SYSTem:LOGGing:STATistics:FILE <string>

<string>	File name

Sets the file in which the loggine data will be stored. The tester creates the file when the test starts. The default directory is

"c:/Program Files/Agilent/RouterTester/data/<SessionType>"

where <SessionType> is type of session (for example, IpPerformance).

If you want to save files to another location, you must specify the full directory path including the drive designator (c:/). Use forward slashes (/), not backslashes (\).

Caution: If you save to an existing log file, the new data replaces the old data.

Statistics are saved in CSV (Comma Separated Values) format. Files should (but are not required to) have this .csv extension. By using this extension, Windows applications that can open or import these files (for example, Microsoft Excel) will be able to recognize them.

The corresponding query returns the current file name.

SYSTem:LOGGing:STATistics:FILE?

Returns:	<string>

SYSTem:LOGGing:STATistics:INTerval <numeric>

<numeric>	PortID, PortID	A List of Port ID’s

Indicates which statistics samplings are saved. Use the default of 600 to save every sampling interval's statistics. If you are saving many statistics and experience performance problems, use a higher number.

The corresponding query returns the current interval value.

SYSTem:LOGGing:STATistics:INTerval?

SYSTem:LOGGing:STATistics:PORTs <string>

<string>	PortID, PortID	A List of Port ID’s

Takes in a list of port ID’s for the test ports for which you wish to log statistics.

The corresponding query return a list of selected ports.

SYSTem:LOGGing:STATistics:PORTs?

Returns:	<string>

SYSTem:LOGGing:STATistics:AVAilable?

Returns a list of the currently available statistics which may be selected for logging.

SYSTem:LOGGing:STATistics:SELect <string>

<string> SONET_SECTION_BITS
SONET_PATH_BITS
SONET_LINE_BITS
SONET_B1_ERROR_COUNT
SONET_B1_ERRORED_SECONDS
SONET_B1_ERROR_RATIO
SONET_B2_ERROR_COUNT
SONET_B2_ERRORED_SECONDS
SONET_B2_ERROR_RATIO
SONET_REIL_ERROR_COUNT
SONET_REIL_ERRORED_SECONDS
SONET_REIL_ERROR_RATIO
SONET_LOS_ERRORED_SECONDS
SONET_LOF_ERRORED_SECONDS
SONET_SEF_ERRORED_SECONDS
SONET_AISL_ERRORED_SECONDS
SONET_RDIL_ERRORED_SECONDS
SONET_B3_ERROR_COUNT
SONET_B3_ERRORED_SECONDS
SONET_B3_ERROR_RATIO
SONET_BIT_ERROR_COUNT
SONET_BIT_ERRORED_SECONDS
SONET_BIT_ERROR_RATIO
SONET_REIP_ERROR_COUNT
SONET_REIP_ERRORED_SECONDS
SONET_REIP_ERROR_RATIO
SONET_AISP_ERRORED_SECONDS
SONET_RDIP_ERRORED_SECONDS
SONET_UNEQP_ERRORED_SECONDS
SONET_PSL_ERRORED_SECONDS
SONET_PDIP_ERRORED_SECONDS
SONET_LOPP_ERRORED_SECONDS
SONET_POINTER_ACTIVITY_ERRORED_SECONDS
SONET_SERVICE_DISRUPTION_LAST_TIME

SONET_SERVICE_ DISRUPTION_COUNT
SONET_SERVICE_DISRUPTION_MAX_TIME
SONET_B1_ERRORED_SECONDS
SONET_B1_ERROR_RATIO

This command accepts a list of statistics as a comma separated list which defines which statistics are to be logged.

The corresponding query returns a common separated list defining which statistics have been selected for logging.

SYSTem:LOGGing:STATistics:SELect?

Returns: <string>

IEEE Commands

*CLS

Clear Status - Clears all status registers and the error queue.

*ESE <numeric>

<numeric> =	1	Operation Complete
	2	Request Control
	4	Query Error
	8	Device Dependent Error
	16	Execution Error
	32	Command Error
	64	User Request
	128	Power On

Event Status Enable - Sets the mask of the Event Status Register.

*ESE?

<numeric> =	0 to 65535	Current Mask

Event Status Enable Query - Returns the current mask setting.

*ESR?

<numeric> =	0 to 65535	Event Status Reg Status

Event Status Register Query - Returns the state of the Event Status Register in numeric form.

*IDN?

<string> =	Comma separated	Manufacturer
		Model number

		Serial Number
		Software Version

Identification Query - Returns the Manufacture Name, Model Number & Name, Serial Number, Software Revision Number as a string. For example:-

"Agilent Technologies, Omniber Xm, GB6533273475, 1.0.0.17" GB signifies the country of origin (Great Britain).

***OPC**

Operation Complete - Masks the OPC bit in the Event Status Register when all pending operations have completed.

***OPC?**

<boolean> =		Status of Operation Complete

Operation Complete Query - Returns a 1 when all pending operations have completed.

***OPT?**

<string> =

Option Identification Query - Returns the Options fitted in the instrument as a comma separated list of option numbers.

***RST**

Set the instrument to its Default settings.

***SRE <numeric>**

<numeric> =	8	QUES Status Summary
	16	Message Available
	32	Event Status Summary
	64	Request Service

128	OPER Status Summary

Service Request Enable - Sets the status byte mask.

***SRE?**

<numeric> = 0 to 65535

Service Request Enable Query - Returns the current mask setting in numeric form.

***STB?**

<numeric> = 0 to 65535

Status Byte Query - Returns the value of the status byte in numeric form.

SDH Command Reference

This chapter contains detailed information on the SCPI (Standard Commands for Programming Instruments) and IEEE 488.2 common commands you will use when writing programs to control your Instrument for SDH operation.

SOURce Commands - Transmitter SDH Settings Commands

:SOURce:DATA:TELecom:SDH:RATE <discrete> <portid>

Controls the characteristics of the instrument’s output ports.

<discrete> =	STM64	10Gb/s
	STM16	2.5Gb/s
	STM4	622Mb/s
	STM1	155Mb/s

Sets the output rate for the instrument output port.

The corresponding query returns the output port rate in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:RATE? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SDH:SCRambling <boolean> <portid>

<boolean> =	0 or OFF	SDH Scrambling Off
	1 or ON	SDH Scrambling On

Enables or Disables SDH Scrambling.

The corresponding query returns the scrambling setting as 0 or 1.

:SOURce:DATA:TELecom:SDH:SCRambling? <portid>

Returns: <boolean>

:SOURce:DATA:TELecom:SDH:CHANnels <string> <portid>

<string> =	Large String	A large string of characters separated by commas.

This command is used to define the Channel Setup for an SDH Signal. It is constructed from the following units, AU-3, AU-4, AU-4-2c, AU-4-3c, AU-4-4c, AU-4-8c, AU-4-16c, AU-4-64c. The range of values for the parameter is as follows

AU3

AU4

AU4_2c

AU4_3c

AU4_4c

AU4_8c

AU4_16c

AU4_64c

An example of a section of such a string may be seen below.

```
“AU3,AU3,AU3,AU4_2c,AU4_16c,AU3”
```

The corresponding query returns a string using similar syntax representing the current structure of the SDH Channel Setup.

:SOURce:DATA:TELecom:SDH:CHANnels? <portid>

Returns:	<string>

:SOURce:DATA:TELecom:SDH:CHANnels:EQUipped <list> <portid>

<list> =	List	A list of integers separated by commas surrounded by brackets. ()

This command is used to define which of the channels specified by the channel list command above are equipped. The <string> consists of a list of integers which indicate the position of the channels to be selected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be equipped. The 75th channel through to the 90th should be equipped and finally the 94th channel should be equipped.

The corresponding query returns a string using similar syntax representing the currently equipped channels.

:SOURce:DATA:TELecom:SDH:CHANnels:EQUipped? <portid>

Returns:	<list>

:SOURce:DATA:TELecom:SDH:CHANnels:UNEQuipped <list> <portid>

<string> =	List	A list of integers separated by commas commas surrounded by brackets. ()

This command is used to define which of the channels specified by the channel list command above are unequipped. The <string> consists of a list of integers which indicate the position of the channels to be unselected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be unequipped. The 75th channel through to the 90th should be unequipped and finally the 94th channel should be unequipped.

:SOURce:DATA:TELecom:SDH:CHANnels:INVert <portid>

This command inverts the currently equipped set of channels.

:SOURce:DATA:TELecom:SDH:CHANnels:ALL <portid>

This command equips all the channels on a given port.

:SOURce:DATA:TELecom:SDH:CHANnels:AT? <numeric> <portid>

	<numeric> =	Channel Position	A LONG indicating the postion of a channel.

This command returns the channel type of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A ChannelType is returned.

returns	<string>	Channel Type	AU-4, AU-4-4C etc.

:SOURce:DATA:TELecom:SDH:CHANnels:SPOSition? <numeric> <portid>

	<numeric> =	Channel Position	A LONG indicating the postion of a channel.

This command returns the start position of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A Start position is then returned.

returns	<numeric>	Long	STS1 Start position.

SOURce Commands - Transmitter SDH Overhead Set Up

:SOURce:DATA:TELecom:SDH:OVERhead:DEFault <portid>

Sets all overhead bytes to their default value:

:SOURce:DATA:TELecom:SDH:OVERhead:DATA <numeric>, <numeric>, <discrete>, <numeric> <portid>

<numeric> =	1 to 64	STM-1 Number
<numeric> =	1 to 3	STM-0 Number
<discrete> =	A1 | A2 | E1 | F1 | D1 | D2 | D3 | K1 | K2_2 | K2 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | B1 | B2 | H1 | H2 | H3 | M1 | E2 | S1 | X11 | X12 | X13 | X21 | X22 | X23 | X31 | X32 | X33 | X41 | X42 | X43 | X51 | X52 | X53 | X61 | X62 | X63 | X71 | X72 | X73 | X81 | X82 | X83 | X91 | X92 | X93 |	
<numeric> =	0 to 255	Byte Value

Sets the value of the selected transmitter section overhead byte. The required byte is specified by 5 command parameters.

The first parameter, STM-1 Number, identifies an STM-1 within the signal. The acceptable range for this parameter will depend on the selected transmit signal rate. At present on the instrument, this is a range of 1 - 64.

The second parameter identifies an STM-0 within the selected STM-1. This may be between 1 – 3.

The third parameter identifies the specific byte in the selected STM-0. There are two ways of specifying this byte. The first is to use standard names where these are valid. The set of valid names is shown in the table above.

The fourth command parameter is the new value that will be transmitted in the specified byte. This value can be specified in hex, octal or decimal format.

The fifth parameter specifies the port.

The corresponding query returns the value set of the byte named within the selected STM-1 column.

:SOURce:DATA:TELecom:SDH:OVERhead:DATA? <numeric>, <numeric> <discrete> <portid>

:SOURce:DATA:TELecom:SDH:OVERhead:J0:PATTern <discrete> <portid>

<discrete> =	B16DEF	16 Byte Default Value
	B16Crc	16 Byte Sequence (with CRC)
	B64DEF	64 Byte Default Value
	B64	64 Byte Sequence

Sets the type of pattern that is to be transmitted in the J0 byte of the STM regenerator section overhead. The pattern repeats every 16 characters and is transmitted character by character in subsequent frames.

B16DEF implies that J0 should contains the 16 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

"Agt ppp<padding><CRC-7>"

‘Agt’ is an abbreviation for ‘Agilent’. Padding is NULL characters

B64DEF implies that J0 should contains the 64 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

“Agilent OmniBER XM Jxxxxx Port ppp<padding><CR><LF>”

Instrument number Jxxxxx will match current instrument. Padding is NULL characters.

The corresponding query returns the type of pattern being transmitted in overhead byte J0 in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:OVERhead:J0:PATTern? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SDH:OVERhead:J0:PATTern:B16Crc <string> <portid>

<string>

Sets the 16-byte sequence of the J0 byte of the regenerator section overhead. The command parameter is a 15 character long string. The instrument automatically appends an E.164 CRC character to make up a 16 character sequence.

If the string is not 15 characters long the instrument will either append NULLS or truncate the string to make it 15 characters long. The pattern repeats every 16 characters and is transmitted character by character in subsequent frames.

Only available when is set to B16Crc.

The corresponding query returns the value of the J0 sequence as a 15-byte string. If the string contains any non-printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SDH:OVERhead:J0:PATTern:B16Crc? <portid>

Returns: <string>

:SOURce:DATA:TELecom:SDH:OVERhead:J0:PATTern:B64 <string> <portid>

Sets the 64-byte sequence of the J0 byte of the regenerator section overhead.

If the string is not 64 characters long the instrument will either append NULLS or truncate the string and terminate with a CR/LF to make it 64 characters long. The pattern repeats every 64 characters and is transmitted character by character in subsequent frames.

Only available when is set to B64.

The corresponding query returns the value of the 64-byte J0 sequence as a string. If the string contains any non-printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SDH:OVERhead:J0:PATTern:B64? <portid>

Returns:	<string>

:SOURce:DATA:TELecom:SDH:POVerhead:DEFault <channel> <portid>

Sets all path overhead bytes to their default value.

:SOURce:DATA:TELecom:SDH:POVerhead:DATA <discrete>,<numeric> <channel> <portid>

<discrete>=	C2 | G1 | F2 | H4 | F3 | K3 | N1	
<numeric>=	0 to 255	Byte Value

Sets the value of the specified VC-4-64c, VC-4-16c, VC-4-4c, VC-4, VC-3 foreground high order path overhead byte.

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the byte specified in numeric form, as described above.

:SOURce:DATA:TELecom:SDH:POVerhead:DATA? <discrete> <channel> <portid>

Returns:	<numeric>

:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern <discrete> <channel> <portid>

<discrete> =	B16DEF	16 Byte Default Value
	B16Crc	16 Byte Sequence (with CRC)
	B64DEF	64 Byte Default Value
	B64	64 Byte Sequence

Sets the type of sequence to be transmitted within the J1 byte of the foreground high order path overhead.

B16DEF implies that J1 should contains the 16 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

"Agt ppp<padding><CRC-7>"

‘Agt’ is an abbreviation for ‘Agilent’. Padding is NULL characters

B64DEF implies that J1 should contains the 64 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

“Agilent OmniBER XM Jxxxxx Port ppp<padding><CR><LF>”

Instrument number Jxxxxx will match current instrument. Padding is NULL characters

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the specified sequence type in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern? <channel> <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern:B16Crc <string> <channel> <portid>

Sets the 16-byte sequence of the J1 byte of the foreground high order path overhead. The command parameter is a 15 characters long string. The instrument automatically appends an E.164 CRC character to make up a 16 character sequence.

If the string is not 15 characters long the instrument will either append NULLS or truncate the string to make it 15 characters long. The pattern repeats every 16 characters and is transmitted character by character in subsequent frames.

Only valid when :SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern is set to B16Crc.

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the value of the J1 sequence as a 15-byte string. If the string contains any non printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern:B16Crc? <channel> <portid>

Returns: <string>

:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern:B64 <string> <channel> <portid>

Sets the 64-byte sequence of the J1 byte of the foreground high order path overhead.

If the string is not 64 characters long the instrument will either append NULLS or truncate the string and terminate with a CR/LF to make it 64 characters long. The pattern repeats every 64 characters and is transmitted character by character in subsequent frames.

Only available when :SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern is set to B64.

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the value of the 64-byte J1 sequence as a string. If the string contains any non-printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern:B64? <channel> <portid>

Returns: <string>

SOURce Commands - Transmitter Error Test Functions

:SOURce:DATA:TELecom:SDH:ERRor:CHANnels:SELected <list> <portid>

<string> =	List	A list of integers separated by commas commas surrounded by brackets. ()

This command is used to select which channels will be tested with the error and alarm functions. Each of the channels defined by the **:SOURce:DATA:TELecom:SDH:CHANnels <string>** command may be either selected or unselected. This command accepts as a parameter, a list of integers which indicate the position of the channels to be selected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th, 56th, 75th 90th and 94th channels should be selected.

The corresponding query returns a list of the selected channels.

:SOURce:DATA:TELecom:SDH:ERRor:CHANnels:SELected ? <portid>

Returns:	<string>

:SOURce:DATA:TELecom:SDH:ERRor:CHANnels:UNSelected <list> <portid>

<string> =	List	A list of integers separated by commas.

This command is used to unselect channels which do not require to be tested with the error and alarm functions. Each of the channels defined by the **:SOURce:DATA:TELecom:SDH:CHANnels <string>** command may be either selected or unselected. This command accepts as a parameter, a list of integers which indicate the position of the channels to be unselected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th, 56th, 75th 90th and 94th channels should be unselected.

The corresponding query returns a list of the unselected channels.

:SOURce:DATA:TELecom:SDH:ERRor:CHANnels:INVert <portid>

This command inverts the current selection of channels which are to be tested for errors and alarms.

:SOURce:DATA:TELecom:SDH:ERRor:CHANnels:ALL <portid>

This command selects all the channels on a given port to be tested for errors and alarms.

:SOURce:DATA:TELecom:SDH:ERRor:GROup <discrete> <portid>

<discrete> =	SECTion	SDH Section Errors
	PATH	SDH Path Errors
	PAYLoad	PAYLoad Errors

Selects SDH transmit test function Error Group. Selection of PATTern results in BIT error type being selected.

The corresponding query returns the Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:ERRor:GROup? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:SDH:ERRor:SECTion <discrete> <portid>

<discrete> =	RSBip	RS-BIP, B1 Errors
	MSBip	MS-BIP, B2 Errors
	MSRei	MS-REI Errors

Selects Section Error Type to generate. The corresponding query returns the Section Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:ERRor:SECTion? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:SDH:ERRor:SECTion:MSRei <discrete> <portid>

<discrete> =	M0M1	Uses M0 & M1
	M1	Uses M1 only

This command allows the user to specify weather a pair of bytes (M0 M1) should be used for errors or if the M1 byte should be used alone. The corresponding query returns MSRei mode currently selected.

:SOURce:DATA:TELecom:SDH:ERRor:SECTion:MSRei? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:SDH:ERRor:PATH <discrete> <portid>

<discrete> =	PBIP	Path Bip, B3 Errors
	HPRei	HP-REI Errors

Selects Path Error Type to generate. The corresponding query returns the Path Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:ERRor:PATH? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:SDH:ERRor:PAYLoad <discrete> <portid>

<discrete> =	BIT

Selects Payload Error Type to generate. The corresponding query returns the Payload Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:ERRor:PAYLoad? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:ERRor:SINGle <portid>

Injects a single error.

There is no corresponding query.

:SOURce:DATA:TELecom:SDH:ERRor:RATE:USER <discrete> <numeric> <discrete> <portid>

<discrete>	SDH_LINE SDH_AU3 SDH_AU4 SDH_AU4_2c SDH_AU4_3c SDH_AU4_4c SDH_AU4_8c SDH_AU4_16c SDH_AU4_64c	SDH Rate Types
<numeric>	1.00 to 9.99	Error Rate Base

<discrete>	1E-3 1E-4 1E-5 1E-6 1E-7 1E-8 1E-9 1E-10	Error Rate Power

Sets the user defined Error Add rate.

```
Note            The maximum user defined error rate is
                dependent on both error type and line
                rate.
```

:SOURce:DATA:TELecom:SDH:ERRor:RATE:USER? <discrete> <portid>

<discrete> =	SDH_LINE SDH_AU3 SDH_AU4 SDH_AU4_2c SDH_AU4_3c SDH_AU4_4c SDH_AU4_8c SDH_AU4_16c SDH_AU4_64c	SDH Rate Types

This query returns the error rate as defined by the user. The Error rate is relative to the Rate type specified and the port.

Returns:	<numeric>	Error Rate Base
	<string>	Error Rate Power

:SOURce:DATA:TELecom:SDH:ERRor:RATE:USER:ACTion <boolean> <portid>

<boolean>=	OFF	
	ON	User Value set as Error Rate

Sets the user defined Error Add rate.

:SOURce:DATA:TELecom:SDH:ERRor:RATE:USER:ACTion:ALL <boolean>

Sets the user defined Error Add rate on all ports.

SOURce Commands - Transmitter Alarm Test Functions

:SOURce:DATA:TELecom:SDH:ALARm:GROup <discrete> <portid>

<discrete> =	SECTion	Section Alarms
	PATH	Path Alarms

Selects Alarm Group.

:SOURce:DATA:TELecom:SDH:ALARm:SECTion ,
:SOURce:DATA:TELecom:SDH:ALARm:PATH …etc,

The corresponding query returns the Alarm Group selected in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:ALARm:GROup? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:SDH:ALARm:SECTion <discrete> <portid>

<discrete> =	LOF	Loss of Frame
	OOF	Out of Frame
	LOS	Loss of Signal Alarm
	MSAis	MS-AIS alarm indication signal
	MSRDi	MS-RDI remote defect indication

Selects Section Alarms.

The corresponding query returns the Section Alarm selected in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:ALARm:SECTion? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SDH:ALARm:OOF <portid>

Sets the OOF alarm Active.There is no corresponding query

:SOURce:DATA:TELecom:SDH:ALARm:PATH <discrete> <portid>

<discrete> =	PAIS	Path AIS
	AULop	AU-LOP Loss of AU pointer
	HPRDi	HP-RDI remote defect indication
	PSL	PSL
	PDIP	PDI-P
	PUNequipped	Path Unequipped

Selects Path Alarms.

The corresponding query returns the Path Alarm selected in discrete form as listed above.

:SOURce:DATA:TELecom:SDH:ALARm:PATH? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SDH:ALARm:PATH:ERDIP:MODE <boolean> <portid>

Sets the Enhanced RDIP Mode on or off for a given port.

The corresponding query reuturns a boolean representing the current ERDIP status.

:SOURce:DATA:TELecom:SDH:ALARm:PATH:ERDIP:MODE? <portid>

:SOURce:DATA:TELecom:SDH:ALARm:PATH:VALue <discrete> <numeric> <portid>

Sets the value for the path alarm. Only Valid for RDI-P & PDI-P.

The correspondig query returns the path alarm value.

:SOURce:DATA:TELecom:SDH:ALARm:PATH:VALue? <discrete> <portid>

:SOURce:DATA:TELecom:SDH:ALARm:PATH:VALue:DEFault <discrete> <portid>

Sets the value for the path alarm to default.

:SOURce:DATA:TELecom:SDH:ALARm:SINGle:ALL

Enables single alarm generation on all ports.

SOURce:DATA:TELecom:SDH:ALARm:TRANsmit <port id>

Transmits a single (SEF/OOF) alarm only if it is pre-configured.

SOURce Commands - Pointer Adjust Test Functions

:SOURce:DATA:TELecom:SDH:POINter:VALue <numeric> <boolean> <portid>

<numeric>	PointerValue	0-782
<boolean>	NDF State	Flag On/Off

A single pointer value is transmitted in all channels. This command may be used to set a new pointer value with or without NDF (New Data Flag).

When the pointer value is moved, it moves simultaneously in all channels at once.

The corresponding query returns the pointer value.

:SOURce:DATA:TELecom:SDH:POINter:VALue? <portid>

Returns:	<numeric>

:SOURce:DATA:TELecom:SDH:POINter:VALue:INCRement <portid>

This command is used to increment the pointer.

:SOURce:DATA:TELecom:SDH:POINter:VALue:DECRement <portid>

This command is used to Decrement the pointer.

SOURce Commands - Frequency Offset Test Functions

:SOURce:CLOCk:SDH:FOFFset:OFFSet <numeric><portid>

<numeric> =	-100.00 to +100.00 ppm

Sets the amount of Clock Frequency Offset.

The corresponding query returns the amount of Clock Frequency Offset in parts per million.

:SOURce:CLOCk:SDH:FOFFset:OFFSet? <portid>

Returns:	<numeric>

SOURce Commands - APS Messages

:SOURce:DATA:TELecom:SDH:OVERhead:APS <numeric> <numeric> <portid>

<numeric> =	LONG	K1 byte
<numeric> =	LONG	K2 byte

Sets the Automatic Protection Switching K1, K2 bytes.

The corresponding query returns the two values, for K1 and K2.

:SOURce:DATA:TELecom:SDH:OVERhead:APS? <portid>

Returns:	<numeric> <numeric>

SOURce Commands - Thru Mode

:SOURce:DATA:TELecom:SDH:THRumode:COVerwrite <discrete> <portid>

<discrete> =	TRANsparent	Transparent.
	INTRusive	Intrusive

Sets the thru mode into either intrusive or transparent mode. When switched on, overhead overwrite is enabled and thru mode is intrusive.

The corresponding query returns the a discrete value for the thru mode state.

:SOURce:DATA:TELecom:SDH:THRumode:COVerwrite? <portid>

Returns:	<discrete>

SENSe Commands - Receiver SDH Settings

:SENSe:DATA:TELecom:SDH:RATE <discrete> <portid>

<discrete> =	STM64	10Gb/s
	STM16	2.5Gb/s
	STM4	622Mb/s
	STM1	155Mb/s

Sets the input rate for the instrument input port.

The corresponding query returns the rate selected in discrete form as listed above.

:SENSe:DATA:TELecom:SDH:RATE? <portid>

Returns: <discrete>

:SENSe:DATA:TELecom:SDH:SCRambling <boolean> <portid>

<boolean> =	0 or OFF	SDH Scrambling Off
	1 or ON	SDH Scrambling On

Enables or Disables SDH Scrambling.

The corresponding query returns the scrambling setting as 0 or 1.

:SENSe:DATA:TELecom:SDH:SCRambling? <portid>

Returns: <boolean>

:SENSe:DATA:TELecom:SDH:CHANnels <string> <portid>

<string> =	Large String	A large string of characters separated by commas.

This command is used to define the Channel Setup for an SDH Signal. It is constructed from the following units, AU-3, AU-4, AU-4-2C, AU-4-3C, AU-4-4C, AU-4-8C, AU-4-16C, AU-4-64C. The range of values for the parameter is as follows

AU3

AU4

AU4_2C

AU4_3C

AU4_4C

AU4_8C

AU4_16C

AU4_64C

An example of a section of such a string may be seen below.

```
“AU3,AU3,AU3,AU4_2C,AU4_16,AU3”
```

The corresponding query returns a string using similar syntax representing the current structure of the SDH Channel Setup.

:SENSe:DATA:TELecom:SDH:CHANnels? <portid>

Returns:	<string>

:SENSe:DATA:TELecom:SDH:CHANnels:EQUipped <list> <portid>

<list> =	List	A list of integers separated by commas surround by brackets.()

This command is used to define which of the channels specified by the channel list command above are equipped. The <string> consists of a list of integers which indicate the position of the channels to be selected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be equipped. The 75th channel through to the 90th should be equipped and finally the 94th channel should be equipped.

The corresponding query returns a string using similar syntax representing the currently equipped channels.

:SENSe:DATA:TELecom:SDH:CHANnels:EQUipped? <portid>

Returns:	<string>

:SENSe:DATA:TELecom:SDH:CHANnels:UNEQuipped <list> <portid>

<string> =	List	A list of integers separated by commas enclosed in brackets ()

This command is used to define which of the channels specified by the channel list command above are unequipped. The <string> consists of a list of integers which indicate the position of the channels to be unselected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be unequipped. The 75th channel through to the 90th should be unequipped and finally the 94th channel should be unequipped.

:SENSe:DATA:TELecom:SDH:CHANnels:INVert <portid>

This command inverts the currently equipped set of channels.

:SENSe:DATA:TELecom:SDH:CHANnels:ALL <portid>

This command equips all the channels on a given port.

:SENSe:DATA:TELecom:SDH:CHANnels:AUTO <portid>

The receiver can automatically detect the mixture of channel types at its input, and configure its channel-settings appropriately.

:SENSe:DATA:TELecom:SDH:CHANnels:UAUTo <portid>

Undo command. This will undo
:SENSe:DATA:TELecom:SDH:CHANnels:AUTO

:SENSe:DATA:TELecom:SDH:CHANnels:AT? <numeric> <portid>

	<numeric> =	Channel Position	A LONG indicating the postion of a channel.

This command returns the channel type of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A ChannelType is returned.

returns	<string>	Channel Type	AU-4, AU-4-4C etc.

:SENSe:DATA:TELecom:SDH:CHANnels:SPOSition? <numeric> <portid>

	<numeric> =	Channel Position	A LONG indicating the postion of a channel.

This command returns the start position of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A Start position is then returned.

returns	<numeric>	Long	STS1 Start position.

:SENSe:DATA:TELecom:SDH:J1:EXPected <discrete> <string> <channel> <portid>

<discrete>=	B16 B64	Length of expected path trace message.
<string> =	STRING	The expected J1 string for the given channel.

Set the expected value for the J1 String. (Path Trace Message)

The corresponding query returns the expected value of the J1 string.

:SENSe:DATA:TELecom:SDH:J1:EXPected? <channel> <portid>

Returns:	<string>

The corresponding query returns the expected length of the J1 string.

:SENSe:DATA:TELecom:SDH:J1:EXPected:LENGth? <channel> <portid>

Returns:	<discrete>

SENSe:DATA:TELecom:SDH:J1:EXPected:SET:RECeived:ALL <porthandle>

This command may be used to set all the expected J1 Trace messages as the recived.

SENSe:DATA:TELecom:SDH:J1:EXPected:SET:RECeived <porthandle> <channel>

This command may be used to set a J1 Trace message on a given channel as the recived.

SENSe:DATA:TELecom:SDH:J1:EXPected:COMPare:STARt <porthandle>

This command starts the Path Trace message check on a given port.

The corresponding query returns a boolean to state wether the path trace is running.

SENSe:DATA:TELecom:SDH:J1:EXPected:COMPare:STARt? <porthandle>

SENSe:DATA:TELecom:SDH:J1:EXPected:MATCh? <porthandle> <channel>

This command returns a boolean to represent the result of a path trace compare against the expected value for a given channel on a given port.

SENSe:DATA:TELecom:SDH:J1:EXPected:MATCh:ALL? <porthandle>

This command does the same as above but for all channels.

:SENSe:DATA:TELecom:SDH:OCAPture:CHANnel:SELect <numeric> <portid>

Sets the channel for over head capture.

The corresponding query returns the selected channel.

:SENSe:DATA:TELecom:SDH:OCAPture:CHANnel:SELect? <portid>

:SENSe:DATA:TELecom:SDH:OCAPture:OHBYte <discrete> <portid>

Sets the byte which is to be captured.

The corresponding query returns the overhead byte selected.

:SENSe:DATA:TELecom:SDH:OCAPture:OHBYte? <portid>

:SENSe:DATA:TELecom:SDH:OCAPture? <portid>

Returns the current capture state as a boolean.

SENSe Commands – SDH SENSe:DATA commands

SDH Total Results

Each of the SENSe:DATA commands may be applied to a given set of ports. The ports must forst be selected using the following command.

In order to select a certain channel or group of channels, then the following command should be used.
:SENSe:DATA:TELecom:SDH:CHANnels:EQUipped <string> <portid>

See above commands for more details.

Returned values for SENSe:DATA

Each of the SENSe:DATA commands will return a set of floating point values which represent the values for each of the channels selected. The data will be returned in the form of a comma separated list.

SDH Error Count Results

:SENSe:DATA? <string> <portid>

<string> = **"ECOunt:SDH:<error>"** <error> is one of the following

RSBip	RS B1 BIP error count
MSBip	MS B2 BIP error count
MSRei	MS REI error count
PBIP	Path B3 BIP error count
HPRei	HP-REI error count
BIT	Bit Error

SDH Error Ratio Results

:SENSe:DATA? <string> <portid>

<string> = **"ERATio:SDH:<error>"** <error> is one of the following

RSBip	RS B1 BIP error ratio
MSBip	MS B2 BIP error ratio
MSRei	MS REI error ratio.
PBIP	Path B3 BIP error ratio
HPRei	HP-REI error ratio
BIT	Bit Error

SDH Pointer Activity Results

:SENSe:DATA? <string> <portid>

<string> = **"PACTivity:SDH:<type>"** where <type> is one of the following

SEConds	AU Pointer Seconds
PCOunt	Pointer Increment Counter
NCOunt	Pointer Decrement Counter

SDH Power Levels

:SENSe:DATA? <string> <portid>

<string> = **"OPOWer:SDH"**

SDH Alarm Seconds Results

:SENSe:DATA? <string> <portid>

<string> = **"ASEConds:SDH:<alarm>"** <alarm> is one of the following

LOF	Loss Of Frame
OOF	Out Of Frame
AULop	Loss Of Pointer
MSAis	Multiplexer Section AIS
PAIS	Path AIS
MSRDi	Multiplexer Section Remote Defect
B1	B1 Erros
B2	B3 Erros
B3	B3 Errors
MSRei	MS-REI
HPRei	HP-REI
HPRDi	HP-RDI
PUNeq	Path Unequipped
PDIP	PDI-P

SDH Analysis Results

G.828 Analysis

:SENSe:DATA? <string> <portid>

<string> = **"<result type> :SDH: <path type> : G828"**

eg. :SENSe:DATA? “ESR:SDH:PATH:G828”, 010101

<result type>	<path type>
ESEConds	CVS
SESeconds	CVL
SEPeriod	REIL

EBCount	CVP
BBECount	REIP
ESRatio	
SESRatio	
SEPI	
BBERatio	
UASeconds	
PUASeconds	

GR.253 Analysis

:SENSe:DATA? <string> <portid>

<string> = **"<result type> :SDH: <path type> : GR253"**

eg. :SENSe:DATA? "ESR:SDH:PATHFG828", 010101

<result type>	<path type>
ESEConds	CVS
SESeconds	CVL
ECOunt	REIL
UASeconds (SEFSeconds for CVS)	CVP
	REIP

SENSe:DATA:TELecom:GR253:ANALysis:ERDip:MODE <boolean>, <portid>

<boolean> =	0 or OFF	Off
	1 or ON	On

Sets the current ERDIP Mode for GR253. The corresponding query returns the current setting.

SENSe:DATA:TELecom:GR253:ANALysis:ERDip:MODE? <portid>

Returns: <boolean>

FETCh SDH Commands

The FETCh subsystem is used to retrieve data directly accumulated by the instrument.

SDH Overhead Bytes

:FETCh:ARRay:DATA:TELecom:SDH:OVERhead:DATA? <numeric> <portid>

<numeric> =	1 – 64	STM-1 Position

Returns the value of the selected section overhead as a comma seperated list of 31 integer numerical values in the range 0 to 255.

Returns: <numeric>, <numeric>,……<numeric>

The values are arranged as shown

<row1 col1>,<row1 col2>,<row1 col3>.. ..<row1 col9>

<row2 col1>,<row2 col2>,<row2 col3>.. ..<row2 col9>

<row3 col1>.............

....<row8 col9>

<row9 col1>,<row9 col2>,<row9 col3>.. ..<row9 col9>

:FETCh:SCALar:DATA:TELecom:SDH:OVERhead:DATA? <numeric> <numeric> <discrete> <portid>

<numeric> =	1 to 64	STM-1
<numeric> =	1 to 3	STM-0 within STM-1
<discrete> =	A1 | A2 | E1 | F1 | D1 | D2 | D3 | K1 | K2_2 | K2 |	

D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 |
B1 | B2 | H1 | H2 | H3 | M1 | E2 | S1 |
X11 | X12 | X13 | X21 | X22 | X23 |
X31 | X32 | X33 | X41 | X42 | X43 |
X51 | X52 | X53 | X61 | X62 | X63 |
X71 | X72 | X73 | X81 | X82 | X83 |
X91 | X92 | X93 |

Returns: <numeric>

Returns the value of a single named byte of the selected transmitter section overhead.

:FETCh:DATA:TELecom:SDH:POVerhead:MONitor:CHANnel <channel> <portid>

This command switches the receiver circuits to look at a paticular channel. It is the overhead from this channel which is returned in the following commands. This command must be used prior to any request for Path Overhead bytes.

The corresponding query returns the channel which is currently being monitored.

:FETCh:DATA:TELecom:SDH:POVerhead:MONitor:CHANnel? <portid>

:FETCh:ARRay:DATA:TELecom:SDH:POVerhead:DATA? <portid>

Returns: <numeric>, <numeric>,……<numeric>

Returns the value of the high order path overhead bytes as an array of 9 numeric values. Each numeric is in the range 0 to 255.The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

:FETCh:SCALar:DATA:TELecom:SDH:POVerhead:DATA? <discrete> <portid>

<discrete>= C2 | G1 | F2 | H4 | F3 | K3 | N1 | B3 | J1

Returns: <numeric>

Returns the value of a single named byte of the selected foreground high order path overhead byte. The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

SDH Labelled Overhead Bytes

:FETCh:SCALar:DATA:TELecom:SDH:OVERhead:K1? <portid>

Returns: <numeric>

Returns the value of the K1 APS signalling overhead byte.

:FETCh:SCALar:DATA:TELecom:SDH:OVERhead:K2? <portid>

Returns: <numeric>

Returns the value of the K2 APS signalling overhead byte.

SDH Overhead Trace Messages

:FETCh:STRing:DATA:TELecom:SDH:J0? <portid>

Returns: <string>

Returns the value of the high order J0 path trace byte as a, 64 ASCII character string if CRC7 is not detected, 15 ASCII character string if CRC7 is detected. If the string contains any non printing characters then ~ is substituted.

:FETCh:STRing:DATA:TELecom:SDH:J1? <portid>

Returns: <string>

Returns the value of the high order J1 path trace byte as a, 64 ASCII character string if CRC7 is not detected, 15 ASCII character string if CRC7 is detected. If the string contains any non printing characters then ~ is substituted. The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

:FETCh:STRing:DATA:TELecom:SDH:J0:LENGth? <portid>

Returns: <numeric> Long

This command returns the length as a numeric value, of the high order J0 section trace byte.

:FETCh:STRing:DATA:TELecom:SDH:J1:LENGth? <portid>

Returns: <numeric> Long

This command returns the length as a numeric value, of the high order J1 path trace byte. The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

:FETCh:ARRay:DATA:TELecom:SDH:OCAPture? <portid>

Returns the captured overhead sequence data. The command returns three values: NumTransitions NumBytesInByteGroup CaptureData

SONet Command Reference

This chapter contains detailed information on the SCPI (Standard Commands for Programming Instruments) and IEEE 488.2 common commands you will use when writing programs to control your Instrument for SONet operation.

SOURce Commands - Transmitter SONet Settings Commands

:SOURce:DATA:TELecom:SONet:RATE <discrete> <portid>

Controls the characteristics of the instrument’s output ports.

<discrete> =	OC192	10Gb/s
	OC48	2.5Gb/s
	OC12	622Mb/s
	OC3	155Mb/s

Sets the output rate for the instrument output port.

The corresponding query returns the output port rate in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:RATE? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:SCRambling <boolean> <portid>

<boolean> =	0 or OFF	SONet Scrambling Off
	1 or ON	SONet Scrambling On

Enables or Disables SONet Scrambling.

The corresponding query returns the scrambling setting as 0 or 1.

:SOURce:DATA:TELecom:SONet:SCRambling? <portid>

Returns: <boolean>

:SOURce:DATA:TELecom:SONet:PAYLoad:SCOM <boolean> <portid>

<boolean> =	0 or OFF	SCOM OFF
	1 or ON	SCOM ON

For STS-1 only: a 'Stuff column overwrite mode' can be enabled. In this mode the payload overwrites the Fixed Stuff columns in the SPE/VC. This mode is provided in order to be compatible with OmniBER which implemented STS-1 in this way.

For AU3, stuff columns cannot be overwritten.

:SOURce:DATA:TELecom:SONet:PAYLoad:SCOM ? <portid>

Returns:	<boolean>

:SOURce:DATA:TELecom:SONet:CHANnels <string> <portid>

<string> =	Large String	A large string of characters separated by commas.

This command is used to define the Channel Setup for an SONet Signal. It is constructed from the following units, STS1, STS3C, STS6C, STS9C, STS12C, STS24C, STS48C, STS192C.

The range of values for the parameter is as follows

STS1

STS3C

STS6C

STS9C

STS12C

STS24C

STS48C

STS192C

An example of a section of such a <string> may be seen below.

“STS1,STS1,STS24C,STS12C,STS6C”

The corresponding query returns a string using similar syntax representing the current structure of the SONet Channel Setup.

`:SOURce:DATA:TELecom:SONet:CHANnels ? <portid>`

Returns:	<string>

`:SOURce:DATA:TELecom:SONet:CHANnels:EQUipped <string> <portid>`

<string> =	List	A list of integers separated by commas enclosed in brackets. For example (1,2,3,4)

This command is used to define which of the channels specified by the channel list command above are equipped. The <string> consists of a list of integers which indicate the position of the channels to be selected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be equipped. The 75th channel through to the 90th should be equipped and finally the 94th channel should be equipped.

The corresponding query returns a string using similar syntax representing the currently equipped channels.

`:SOURce:DATA:TELecom:SONet:CHANnels:EQUipped ? <portid>`

Returns:	<string>

:SOURce:DATA:TELecom:SONet:CHANnels:UNEQuipped <string> <portid>

<string> =	List	A list of integers separated by commas enclosed in brackets. For example (1,2,3,4)

This command is used to define which of the channels specified by the channel list command above are unequipped. The <string> consists of a list of integers which indicate the position of the channels to be unselected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be unequipped. The 75th channel through to the 90th should be unequipped and finally the 94th channel should be unequipped.

:SOURce:DATA:TELecom:SONet:CHANnels:INVert <portid>

This command inverts the currently equipped set of channels.

:SOURce:DATA:TELecom:SONet:CHANnels:ALL <portid>

This command equips all the channels on a given port.

:SOURce:DATA:TELecom:SONet:CHANnels:AT? <numeric> <portid>

<numeric> =	Channel Position	A LONG indicating the postion of a channel.

This command returns the channel type of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A ChannelType is returned.

returns	<string>	Channel Type	STS1 STS24C STS12C etc.

:SOURce:DATA:TELecom:SONet:CHANnels:SPOSition? <numeric> <portid>

	<numeric> =	Channel Position	A LONG indicating the postion of a channel.

This command returns the start position of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A Start position is then returned.

returns	<numeric>	Long	STS1 Start position.

SOURce Commands - Transmitter SONet Overhead Set Up

:SOURce:DATA:TELecom:SONet:OVERhead:DEFault <portid>

Sets all overhead bytes to their default value:

:SOURce:DATA:TELecom:SONet:OVERhead:DATA <numeric>, <numeric>, <discrete>, <numeric> <portid>

<numeric> =	1 to 64	STS-3 Number
<numeric> =	1 to 3	STS-1 Number
<discrete> =	A1 | A2 | E1 | F1 | D1 | D2 | D3 | K1 | K2_2 | K2 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | B1 | B2 | H1 | H2 | H3 | M1 | E2 | Z0 | Z1 | Z2 | S1 | X11 | X12 | X13 | X21 | X22 | X23 | X31 | X32 | X33 | X41 | X42 | X43 | X51 | X52 | X53 | X61 | X62 | X63 | X71 | X72 | X73 | X81 | X82 | X83 | X91 | X92 | X93 |	
<numeric> =	0 to 255	Byte Value

Sets the value of the selected transmitter section overhead byte. The required byte is specified by 5 command parameters.

The first parameter, STS-3 Number, identifies an STS-3 within the signal. The acceptable range for this parameter will depend on the selected transmit signal rate. At present on the instrument, this is a range of 1 - 64.

The second parameter identifies an STS-1 within the selected STS-3. This may be between 1 – 3.

The third parameter identifies the specific byte in the selected STS-1. There are two ways of specifying this byte. The first is to use standard names where these are valid. The set of valid names is shown in the table above.

The fourth command parameter is the new value that will be transmitted in the specified byte. This value can be specified in hex, octal or decimal format.

The fifth parameter specifies the port.

The corresponding query returns the value set of the byte named within the selected STS-3 column.

:SOURce:DATA:TELecom:SONet:OVERhead:DATA? <numeric>, <numeric> <discrete> <portid>

:SOURce:DATA:TELecom:SONet:OVERhead:J0:PATTern <discrete> <portid>

<discrete> =	B16DEF	16 Byte Default Value
	B16Crc	16 Byte Sequence (with CRC)
	B64DEF	64 Byte Default Value
	B64	64 Byte Sequence

Sets the type of pattern that is to be transmitted in the J0 byte of the STS regenerator section overhead. The pattern repeats every 16 characters and is transmitted character by character in subsequent frames.

B16DEF implies that J0 should contains the 16 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

"Agt ppp<padding><CRC-7>"

‘Agt’ is an abbreviation for ‘Agilent’. Padding is NULL characters

B64DEF implies that J0 should contains the 64 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

“Agilent OmniBER XM Jxxxxx Port ppp<padding><CR><LF>”

Instrument number Jxxxxx will match current instrument. Padding is NULL characters.

The corresponding query returns the type of pattern being transmitted in overhead byte J0 in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:OVERhead:J0:PATTern? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:OVERhead:J0:PATTern:B16Crc <string> <portid>

Sets the 16-byte sequence of the J0 byte of the regenerator section overhead. The command parameter is a 15 character long string. The instrument automatically appends an E.164 CRC character to make up a 16 character sequence.

If the string is not 15 characters long the instrument will either append NULLS or truncate the string to make it 15 characters long. The pattern repeats every 16 characters and is transmitted character by character in subsequent frames.

Only available when is set to B16Crc.

The corresponding query returns the value of the J0 sequence as a 15-byte string. If the string contains any non-printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SONet:OVERhead:J0:PATTern:B16Crc? <portid>

Returns:	<string>

:SOURce:DATA:TELecom:SONet:OVERhead:J0:PATTern:B64 <string> <portid>

Sets the 64-byte sequence of the J0 byte of the regenerator section overhead.

If the string is not 64 characters long the instrument will either append NULLS or truncate the string and terminate with a CR/LF to make it 64 characters long. The pattern repeats every 64 characters and is transmitted character by character in subsequent frames.

Only available when is set to B64.

The corresponding query returns the value of the 64-byte J0 sequence as a string. If the string contains any non-printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SONet:OVERhead:J0:PATTern:B64? <portid>

Returns:	<string>

:SOURce:DATA:TELecom:SONet:POVerhead:DEFault

Sets all path overhead bytes to their default value.

:SOURce:DATA:TELecom:SONet:POVerhead:DATA <discrete>,<numeric> <channel> <portid>

<discrete>=	C2 | G1 | F2 | H4 | Z3 | Z4 | N1	
<numeric>=	0 to 255	Byte Value

Sets the value of the specified channel foreground high order path overhead byte.

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the byte specified in numeric form, as described above.

:SOURce:DATA:TELecom:SONet:POVerhead:DATA? <discrete> <channel> <portid>

Returns: <numeric>

:SOURce:DATA:TELecom:SONet:POVerhead:J1:PATTern <discrete> <channel> <portid>

<discrete> =	B16DEF	16 Byte Default Value
	B16Crc	16 Byte Sequence (with CRC)
	B64DEF	64 Byte Default Value
	B64	64 Byte Sequence

Sets the type of sequence to be transmitted within the J1 byte of the foreground high order path overhead.

B16DEF implies that J1 should contains the 16 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

"Agt ppp<padding><CRC-7>"

‘Agt’ is an abbreviation for ‘Agilent’. Padding is NULL characters

B64DEF implies that J1 should contains the 64 Byte Default Trace Message. The default trace message identifies the originating port, according to the scheme in section XX.

“Agilent OmniBER XM Jxxxxx Port ppp<padding><CR><LF>”

Instrument number Jxxxxx will match current instrument. Padding is NULL characters

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the specified sequence type in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:POVerhead:J1:PATTern? <channel> <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:POVerhead:J1:PATTern:B16Crc <string> <channel> <portid>

Sets the 16-byte sequence of the J1 byte of the foreground high order path overhead. The command parameter is a 15 characters long string. The instrument automatically appends an E.164 CRC character to make up a 16 character sequence.

If the string is not 15 characters long the instrument will either append NULLS or truncate the string to make it 15 characters long. The pattern repeats every 16 characters and is transmitted character by character in subsequent frames.

Only valid when **:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern** is set to B16Crc.

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the value of the J1 sequence as a 15-byte string. If the string contains any non printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SONet:POVerhead:J1:PATTern:B16Crc? <channel> <portid>

Returns: <string>

:SOURce:DATA:TELecom:SONet:POVerhead:J1:PATTern:B64 <string> <channel> <portid>

Sets the 64-byte sequence of the J1 byte of the foreground high order path overhead.

If the string is not 64 characters long the instrument will either append NULLS or truncate the string and terminate with a CR/LF to make it 64 characters long. The pattern repeats every 64 characters and is transmitted character by character in subsequent frames.

Only available when
:SOURce:DATA:TELecom:SDH:POVerhead:J1:PATTern is set to B64.

This command is channel specific. It takes as a parameter a LONG data type which indicates the position of the channel which the command is to apply to.

The corresponding query returns the value of the 64-byte J1 sequence as a string. If the string contains any non-printing characters, ~ is substituted.

:SOURce:DATA:TELecom:SONet:POVerhead:J1:PATTern:B64? <channel> <portid>

Returns: <string>

SOURce Commands - Transmitter Error Test Functions

:SOURce:DATA:TELecom:SONet:ERRor:CHANnels:SELected <string> <portid>

<string> =	List	A list of integers separated by commas enclosed in brackets. For example (1,2,3,4)

This command is used to select which channels will be tested with the error and alarm functions. Each of the channels defined by the **:SOURce:DATA:TELecom:SONet:CHANnels <string>** command may be either selected or unselected. This command accepts as a parameter, a list of integers which indicate the position of the channels to be selected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th, 56th, 75th 90th and 94th channels should be selected.

The corresponding query returns a list of the selected channels.

:SOURce:DATA:TELecom:SONet:ERRor:CHANnels:SELected? <portid>

Returns:	<string>

:SOURce:DATA:TELecom:SONet:ERRor:CHANnels:UNSelected <string> <portid>

<string> =	List	A list of integers separated by commas enclosed in brackets. For example (1,2,3,4)

This command is used to unselect channels which do not require to be tested with the error and alarm functions. Each of the channels defined by the **:SOURce:DATA:TELecom:SONet:CHANnels <string>** command may be either selected or unselected. This command accepts as a parameter, a list of integers which indicate the position of the channels to be unselected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th, 56th, 75th 90th and 94th channels should be unselected.

The corresponding query returns a list of the unselected channels.

:SOURce:DATA:TELecom:SONet:ERRor:CHANnels:UNSelected? <portid>

Returns: <string>

:SOURce:DATA:TELecom:SONet:ERRor:CHANnels:INVert <portid>

This command inverts the current selection of channels which are to be tested for errors and alarms.

:SOURce:DATA:TELecom:SONet:ERRor:CHANnels:ALL <portid>

This command selects all the channels on a given port to be tested for errors and alarms.

:SOURce:DATA:TELecom:SONet:ERRor:GROup <discrete> <portid>

<discrete> =	SECTion	SONet Section Errors
	PATH	SONet Path Errors
	PAYLoad	PAYLoad Errors

Selects SONet transmit test function Error Group. Selection of PATTern results in BIT error type being selected.

The corresponding query returns the Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:ERRor:GROup? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:ERRor:SECTion <discrete> <portid>

<discrete> =	CVS	CV-S, B1 Errors
	CVL	CV-L, B2 Errors
	REIL	REI-L Errors

Selects Section Error Type to generate. The corresponding query returns the Section Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:ERRor:SECTion? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:ERRor:SECTion:REIL <discrete> <portid>

<discrete> =	M0M1	Uses M0 & M1
	M1	Uses M1 only

This command allows the user to specify whether a pair of bytes (M0 M1) should be used for errors or if the M1 byte should be used alone. The corresponding query returns MSRei mode currently selected.

:SOURce:DATA:TELecom:SONet:ERRor:SECTion:REIL? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:ERRor:PATH <discrete> <portid>

<discrete> =	CVP	Path Bip, B3 Errors

REIP	REI-PErrors

Selects Path Error Type to generate. The corresponding query returns the Path Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:ERRor:PATH? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:SONet:ERRor:PAYLoad <discrete>

<discrete> =	BIT

Selects Payload Error Type to generate. The corresponding query returns the Payload Error Type in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:ERRor:PAYLoad?

Returns:	<discrete>

:SOURce:DATA:TELecom:ERRor:SINGle <portid>

Injects a single error.

There is no corresponding query.

:SOURce:DATA:TELecom:SONet:ERRor:RATE:USER <discrete> <numeric> <discrete> <portid>

<discrete>	SONet_LINE SONet_STS1 SONet_STS3c SONet_STS6c SONet_STS9c SONet_STS12c SONet_STS24c SONet_STS48c SONet_STS192c	SONet Rate Types
<numeric>	1.00 to 9.99	Error Rate Base
<discrete>	1E-3 1E-4 1E-5 1E-6 1E-7 1E-8 1E-9 1E-10	Error Rate Power

Sets the user defined Error Add rate.

Note	The maximum user defined error rate is dependent on both error type and line rate.

:SOURce:DATA:TELecom:SONet:ERRor:RATE:USER? <discrete> <portid>

<discrete> =	SONet_LINE SONet_STS1 SONet_STS3c SONet_STS6c SONet_STS9c SONet_STS12c SONet_STS24c SONet_STS48c SONet_STS192c	SONet Rate Types

This query returns the error rate as defined by the user. The Error rate is relative to the Rate type specified and the port.

Returns:	<numeric>	Error Rate Base
	<string>	Error Rate Power

:SOURce:DATA:TELecom:SONet:ERRor:RATE:USER:ACTion <boolean> <portid>

<boolean>=	OFF	
	ON	User Value set as Error Rate

Sets the user defined Error Add rate.

:SOURce:DATA:TELecom:SONet:ERRor:RATE:USER:ACTion:ALL <boolean>

Sets the user defined Error Add rate on all ports.

SOURce Commands - Transmitter Alarm Test Functions

:SOURce:DATA:TELecom:SONet:ALARm:GROup <discrete> <portid>

<discrete> =	SECTion	Section Alarms
	PATH	Path Alarms

Selects Alarm Group.

:SOURce:DATA:TELecom:SDH:ALARm:SECTion ,
:SOURce:DATA:TELecom:SDH:ALARm:PATH …etc,

The corresponding query returns the Alarm Group selected in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:ALARm:GROup? <portid>

Returns:	<discrete>

:SOURce:DATA:TELecom:SONet:ALARm:SECTion <discrete> <portid>

<discrete> =	LOF	Loss of Frame
	SEF	Severely Errored Frame
	LOS	Loss of Signal Alarm
	AISL	MS-AIS alarm indication signal
	RDIL	MS-RDI remote defect indication

Selects Section Alarms.

The corresponding query returns the Section Alarm selected in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:ALARm:SECTion? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:ALARm:SEF <portid>

Sets the SEF alarm Active. There is no corresponding query

:SOURce:DATA:TELecom:SONet:ALARm:PATH <discrete> <portid>

<discrete> =	AISP	Path AIS
	LOPP	AU-LOP Loss of AU pointer
	RDIP	HP-RDI remote defect indication
	PSL	PSL
	PDIP	PDI-P
	PUNequipped	Path Unequipped

Selects Path Alarms.

The corresponding query returns the Path Alarm selected in discrete form as listed above.

:SOURce:DATA:TELecom:SONet:ALARm:PATH? <portid>

Returns: <discrete>

:SOURce:DATA:TELecom:SONet:ALARm:PATH:ERDIP:MODE <boolean> <portid>

Sets the Enhanced RDIP Mode on or off for a given port.

The corresponding query reuturns a boolean representing the current ERDIP status.

:SOURce:DATA:TELecom:SONet:ALARm:PATH:ERDIP:MODE? <portid>

:SOURce:DATA:TELecom:SONet:ALARm:PATH:VALue <discrete> <numeric> <portid>

Sets the value for the path alarm. Only Valid for RDI-P & PDI-P.

The correspondig query returns the path alarm value.

:SOURce:DATA:TELecom:SONet:ALARm:PATH:VALue? <discrete> <portid>

:SOURce:DATA:TELecom:SONet:ALARm:PATH:VALue:DEFault <discrete> <portid>

Sets the value for the path alarm to default.

:SOURce:DATA:TELecom:SONet:ALARm:SINGle:ALL

Enables single alarm generation on all ports.

SOURce:DATA:TELecom:SONet:ALARm:TRANsmit <port id>

Transmits a single (SEF/OOF) alarm only if it is pre-configured.

SOURce Commands - Pointer Adjust Test Functions

:SOURce:DATA:TELecom:SONet:POINter:VALue <numeric> <boolean> <portid>

<numeric>	PointerValue	0-782
<boolean>	NDF State	Flag On/Off

A single pointer value is transmitted in all channels. This command may be used to set a new pointer value with or without NDF (New Data Flag).

When the pointer value is moved, it moves simultaneously in all channels at once.

The corresponding query returns the pointer value.

:SOURce:DATA:TELecom:SONet:POINter:VALue? <portid>

Returns:	<numeric>

:SOURce:DATA:TELecom:SONet:POINter:VALue:INCRement <portid>

This command is used to increment the pointer.

:SOURce:DATA:TELecom:SONet:POINter:VALue:DECRement <portid>

This command is used to Decrement the pointer.

SOURce Commands - Frequency Offset Test Functions

:SOURce:CLOCk:SONet:FOFFset:OFFSet <numeric><portid>

<numeric> = -100.00 to +100.00 ppm

Sets the amount of Clock Frequency Offset when Frequency Offset is enabled. <boolean> to ON.

The corresponding query returns the amount of Clock Frequency Offset in parts per million.

:SOURce:CLOCk:SONet:FOFFset:OFFSet? <portid>

Returns: <numeric>

SOURce Commands - APS Messages

:SOURce:DATA:TELecom:SONet:OVERhead:APS <numeric> <numeric> <portid>

<numeric> =	LONG	K1 byte
<numeric> =	LONG	K2 byte

Sets the Automatic Protection Switching K1, K2 bytes.

The corresponding query returns the two values, for K1 and K2.

:SOURce:DATA:TELecom:SONet:OVERhead:APS? <portid>

Returns: <numeric>
<numeric>

SOURce Commands – Thru Mode

:SOURce:DATA:TELecom:SONet:THRumode:COVerwrite <boolean> <portid>

<boolean> =	On, Off	Intrusive, Transparent.

Sets the thru mode into either intrusive or transparent mode. When switched on, overhead overwrite is enabled and thru mode is intrusive.

The corresponding query returns a boolean for the thru mode state.

:SOURce:DATA:TELecom:SONet:THRumode:COVerwrite? <portid>

Returns:	<boolean>

SENSe Commands - Receiver SONet Settings

:SENSe:DATA:TELecom:SONet:RATE <discrete> <portid>

<discrete> =	OC192	10Gb/s
	OC48	2.5Gb/s
	OC12	622Mb/s
	OC3	155Mb/s

Sets the input rate for the instrument input port.

The corresponding query returns the rate selected in discrete form as listed above.

:SENSe:DATA:TELecom:SONet:RATE? <portid>

Returns:	<discrete>

:SENSe:DATA:TELecom:SONet:SCRambling <boolean> <portid>

<boolean> =	0 or OFF	SONet Scrambling Off
	1 or ON	SONet Scrambling On

Enables or Disables SONet Scrambling.

The corresponding query returns the scrambling setting as 0 or 1.

:SENSe:DATA:TELecom:SONet:SCRambling? <portid>

Returns:	<boolean>

:SENSe:DATA:TELecom:SONet:PAYLoad:SCOM <boolean> <portid>

<boolean> =	0 or OFF	SCOM OFF

1 or ON	SCOM ON

For STS-1 only: a ‘Stuff column overwrite mode’ can be enabled. In this mode the payload overwrites the Fixed Stuff columns in the SPE/VC. This mode is provided in order to be compatible with OmniBER which implemented STS-1 in this way.

For AU3, stuff columns cannot be overwritten.

:SENSe:DATA:TELecom:SONet:PAYLoad:SCOM ? <portid>

Returns:	<boolean>

:SENSe:DATA:TELecom:SONet:CHANnels <string> <portid>

<string> =	Large String	A large string of characters separated by commas.

This command is used to define the Channel Setup for an SONet Signal. It is constructed from the following units, STS1, STS3C, STS6C, STS9C, STS12C, STS24C, STS48C, STS192C.

The range of values for the parameter is as follows

STS1

STS3C

STS6C

STS9C

STS12C

STS24C

STS48C

STS192C

An example of a section of such a string may be seen below.

“STS1,STS1,STS24C,STS12C,STS6C”

The corresponding query returns a string using similar syntax representing the current structure of the SONet Channel Setup.

`:SENSe:DATA:TELecom:SONet:CHANnels ? <portid>`

Returns:	<string>

`:SENSe:DATA:TELecom:SONet:CHANnels:EQUipped <string> <portid>`

<string> =	List	A list of integers separated by commas enclosed in brackets. For example (1,2,3,4)

This command is used to define which of the channels specified by the channel list command above are equipped. The <string> consists of a list of integers which indicate the position of the channels to be selected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be equipped. The 75th channel through to the 90th should be equipped and finally the 94th channel should be equipped.

The corresponding query returns a string using similar syntax representing the currently equipped channels.

`:SENSe:DATA:TELecom:SONet:CHANnels:EQUipped ? <portid>`

Returns:	<string>

:SENSe:DATA:TELecom:SONet:CHANnels:UNEQuipped <string> <portid>

<string> =	List	A list of integers separated by commas enclosed in brackets. For example (1,2,3,4)

This command is used to define which of the channels specified by the channel list command above are unequipped. The <string> consists of a list of integers which indicate the position of the channels to be unselected. For example (1,4,56,75,90,94) would indicate that the 1st, 4th and 56th channels should be unequipped. The 75th channel through to the 90th should be unequipped and finally the 94th channel should be unequipped.

:SENSe:DATA:TELecom:SONet:CHANnels:INVert <portid>

This command inverts the currently equipped set of channels.

:SENSe:DATA:TELecom:SONet:CHANnels:ALL <portid>

This command equips all the channels on a given port.

:SENSe:DATA:TELecom:SONet:CHANnels:AUTO <portid>

The receiver can automatically detect the mixture of channel types at its input, and configure its channel-settings appropriately.

:SENSe:DATA:TELecom:SONet:CHANnels:UAUTo <portid>

Undo command. This will undo
:SENSe:DATA:TELecom:SONet:CHANnels:AUTO

:SENSe:DATA:TELecom:SONet:CHANnels:AT? <numeric> <portid>

<numeric> =	Channel Position	A LONG indicating the position of a channel.

This command returns the channel type of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A ChannelType is returned.

returns	<string>	Channel Type	STS1 STS24C STS12C etc.

:SENSe:DATA:TELecom:SONet:CHANnels:SPOSition? <numeric> <portid>

<numeric> =	Channel Position	A LONG indicating the position of a channel.

This command returns the start position of a channel at a paticular position. The command takes a port id and a numeric value indicating the position of the channel. A Start position is then returned.

returns	<numeric>	Long	STS1 Start position.

:SENSe:DATA:TELecom:SONet:J1:EXPected <discrete> <string> <channel> <portid>

<discrete>=	16B 64B	Length of expected path trace message.
<string> =	STRING	The expected J1 string for the given channel.

Set the expected value for the J1 String. (Path Trace Message)

The corresponding query returns the expected value of the J1 string.

:SENSe:DATA:TELecom:SONet:J1:EXPected? <channel> <portid>

Returns:	<string>

The corresponding query returns the expected length of the J1 string.

:SENSe:DATA:TELecom:SONet:J1:EXPected:LENGth? <channel> <portid>

Returns: <discrete>

SENSe:DATA:TELecom:SONet:J1:EXPected:SET:RECeived:ALL <porthandle>

This command may be used to set all the expected J1 Trace messages as the recived.

SENSe:DATA:TELecom:SONet:J1:EXPected:SET:RECeived <porthandle> <channel>

This command may be used to set a J1 Trace message on a given channel as the recived.

SENSe:DATA:TELecom:SONet:J1:EXPected:COMPare:STARt <porthandle>

This command starts the Path Trace message check on a given port.

The corresponding query returns a boolean to state wether the path trace is running.

SENSe:DATA:TELecom:SONet:J1:EXPected:COMPare:STARt? <porthandle>

SENSe:DATA:TELecom:SONet:J1:EXPected:MATCh? <porthandle> <channel>

This command returns a boolean to represent the result of a path trace compare against the expected value for a given channel on a given port.

SENSe:DATA:TELecom:SONet:J1:EXPected:MATCh:ALL? <porthandle>

This command does the same as above but for all channels.

:SENSe:DATA:TELecom:SONet:OCAPture:STS1 <numeric> <portid>

Sets the STS1 postion for over head capture.

The corresponding query returns the selected STS1 position.

:SENSe:DATA:TELecom:SONet:OCAPture:STS1? <portid>

:SENSe:DATA:TELecom:SONet:OCAPture:STS3:SELect <numeric> <portid>

Sets the STS3 postion for over head capture.

The corresponding query returns the selected STS3 position.

:SENSe:DATA:TELecom:SONet:OCAPture:STS3:SELect? <portid>

:SENSe:DATA:TELecom:SONet:OCAPture:OHBYte <discrete> <portid>

Sets the byte which is to be captured.

The corresponding query returns the overhead byte selected.

:SENSe:DATA:TELecom:SONet:OCAPture:OHBYte? <portid>

:SENSe:DATA:TELecom:SONet:OCAPture? <portid>

Returns the current capture state as a boolean.

SENSe Commands – Result Returning Commands

SONet Total Results

Each of the SENSe:DATA commands may be applied to a given port id.

In order to select a certain channel or group of channels, then the following command should be used.

:SENSe:DATA:TELecom:SONet:CHANnels:EQUipped <string> <portid>

See above commands for more details.

Returned values for SENSe:DATA

Each of the SENSe:DATA commands will return a set of floating point values which represent the values for each of the channels selected. The data will be returned in the form of a comma separated list.

Sonet Error Count Results

:SENSe:DATA? <string> <portid>

<string> = **"ECOunt:SONet:<error>"** <error> is one of the following

CVS	RS B1 BIP error count
CVL	MS B2 BIP error count
REIL	REI-L error count
CVP	Path B3 BIP error count
REIP	REI-P error count
BIT	Bit Error

SONet Error Ratio Results

:SENSe:DATA? <string> <portid>

<string> = **"ERATio:SONet:<error>"** <error> is one of the following

CVS	RS B1 BIP error ratio
CVL	MS B2 BIP error ratio
REIL	REI-L error ratio
CVP	Path B3 BIP error ratio
REIP	REI-P error ratio
BIT	Bit Error

SONet Pointer Activity Results

:SENSe:DATA? <string> <portid>

<string> = **"PACTivity:SONet:<type>"** where <type> is one of the following

SEConds	AU Pointer Seconds
PCOunt	Pointer Increment Counter
NCOunt	Pointer Decrement Counter

SONet Power Levels

:SENSe:DATA? <string> <portid>

<string> = **"OPOWer:SONet"**

SONet Alarm Seconds Results

:SENSe:DATA? <string> <portid>

<string> = **"ASEConds:SONet:<alarm>"** <alarm> is one of the following

LOF	Loss Of Frame
SEF	Severly Errored Frame
LOPP	Loss Of Pointer
AISL	AIS-L
AISP	AIS-P
RDIL	RDIL
B1	B1 Errors
B2	B3 Errors
B3	B3 Errors
REIL	REI-L
REIP	REI-P
RDIP	RDI-P
BIT	Bit Errors
PUNeq	Path Unequipped
PDIP	PDI-P

G.828 Analysis

:SENSe:DATA? <string> <portid>

<string> = **"<result type> :SONet: <path type> : G828"**

e.g. :SENSe:DATA? “ESR:SONet:PATH:G828”, 010101

<result type>	**<path type>**
ESECconds	CVS
SESeconds	CVL
SEPeriod	REIL
EBCount	CVP
BBECount	REIP
ESRatio	
SESRatio	
SEPI	
BBERatio	
UASeconds	
PUASeconds	

GR.253 Analysis

:SENSe:DATA? <string> <portid>

<string> = **"<result type> :SONet: <path type> : GR253"**

e.g. :SENSe:DATA? “ESR:SONet:PATHFG828”, 010101

<result type>	**<path type>**
ESECconds	CVS
SESeconds	CVL
ECOunt	REIL
UASeconds (SEFSeconds for CVS)	CVP
	REIP

FETCh SONet Commands

The FETCh subsystem is used to retrieve data directly accumulated by the instrument.

SONet Overhead Bytes

:FETCh:ARRay:DATA:TELecom:SONet:OVERhead:DATA? <numeric> <portid>

<numeric> =	1 – 64	STS-3 Position

Returns the value of the selected section overhead as a comma separated list of 27 integer numerical values in the range 0 to 255.

Returns: <numeric>, <numeric>,……<numeric>

The values are arranged as shown

<row1 col1>,<row1 col2>,<row1 col3>.. ..<row1 col9>

<row2 col1>,<row2 col2>,<row2 col3>.. ..<row2 col9>

<row3 col1>….………..

……………… ..<row8 col9>

<row9 col1>,<row9 col2>,<row9 col3>.. ..<row9 col9>

:FETCh:SCALar:DATA:TELecom:SONet:OVERhead:DATA? <numeric> <numeric> <discrete> <portid>

<numeric> =	1 to 64	STS-3
<numeric> =	1 to 3	STS-1 within STS-3
<discrete> =	A1 | A2 | E1 | F1 | D1 | D2 | D3 | K1 | K2_2 | K2 | D4 | D5 | D6 | D7 | D8 | D9 | D10 | D11 | D12 | B1 | B2 | H1 | H2 | H3 | M1 | E2 | Z0 | Z1 | Z2 | S1 |	

X11 | X12 | X13 | X21 | X22 | X23 |
X31 | X32 | X33 | X41 | X42 | X43 |
X51 | X52 | X53 | X61 | X62 | X63 |
X71 | X72 | X73 | X81 | X82 | X83 |
X91 | X92 | X93 |

Returns the value of a single named byte of the selected transmitter section overhead. The required byte is specified by 2 command parameters.

The second parameter identifies the specific byte in the selected set of columns.

:FETCh:DATA:TELecom:SONet:POVerhead:MONitor:CHANnel <channel> <portid>

This command switches the receiver circuits to look at a particular channel. It is the overhead from this channel which is returned in the following commands.

The corresponding query returns the channel which is currently being monitored.

:FETCh:DATA:TELecom:SONet:POVerhead:MONitor:CHANnel? <portid>

:FETCh:ARRay:DATA:TELecom:SONet:POVerhead:DATA? <portid>

Returns: <numeric>, <numeric>,……<numeric>

Returns the value of the high order path overhead bytes as an array of 9 numeric values. Each numeric is in the range 0 to 255.

The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

:FETCh:SCALar:DATA:TELecom:SONet:POVerhead:DATA? <discrete> <portid>

<discrete>= J1 | B3 | C2 | G1 | F2 | H4 | Z3 | Z4 | N1

Returns: <numeric>

Returns the value of a single named byte of the selected foreground high order path overhead byte. The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

:FETCh:ARRay:DATA:TELecom:SONet:OCAPture? <portid>

Returns the captured overhead sequence data. The command returns three values: NumTransitions NumBytesInByteGroup CaptureData

SONet Labelled Overhead Bytes

:FETCh:SCALar:DATA:TELecom:SONet:OVERhead:K1? <portid>

Returns: <numeric>

Returns the value of the K1 APS signalling overhead byte.

:FETCh:SCALar:DATA:TELecom:SONet:OVERhead:K2? <portid>

Returns: <numeric>

Returns the value of the K2 APS signalling overhead byte.

SONet Overhead Trace Messages

:FETCh:STRing:DATA:TELecom:SONet:J0? <portid>

Returns: <string>

Returns the value of the high order J0 path trace byte as a, 64 ASCII character string if CRC7 is not detected, 15 ASCII character string if CRC7 is detected. If the string contains any non-printing characters then ~ is substituted.

:FETCh:STRing:DATA:TELecom:SONet:J1? <portid>

Returns: <string>

Returns the value of the high order J1 path trace byte as a, 64 ASCII character string if CRC7 is not detected, 15 ASCII character string if CRC7 is detected. If the string contains any non-printing characters then ~ is substituted. The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

:FETCh:STRing:DATA:TELecom:SONet:J0:LENGth? <portid>

Returns: <numeric> Long

This command returns the length as a numeric value, of the high order J0 section trace byte.

:FETCh:STRing:DATA:TELecom:SONet:J1:LENGth? <portid>

Returns: <numeric> Long

This command returns the length as a numeric value, of the high order J1 path trace byte.The specific channel which is to be examined must be specified by using the **POVerhead:MONitor:CHANnel** command prior to using this function.

STATus Commands

Common Registers

This Commands controls the status reporting registers. SCPI defined status registers QUEStionable, OPERation and INSTrument are provided.

For each of the **<Status Registers>**’s listed, the following commands exist:

```
:STATus:<Status Register>:ENABle <numeric>
```

Sets the Event Enable register mask which allows true conditions in the Event register to be reported in the **<Status Register>**’s summary bit. If a bit is 1 in the Event Enable register and its associated event bit makes the transition to true, a positive transition will occur in the **<Status Register>**’s summary bit.

The corresponding query returns the current mask setting.

```
:STATus:<Status Register>:ENABle?
```

Returns: <numeric>

```
:STATus:<Status Register>:PTRansition <numeric>
```

Sets the positive Transition Filter. Setting a bit in the positive Transition filter shall cause a 0 to 1 transition in the corresponding bit of the **<Status Register>**’s Condition register to cause a 1 to be written in the corresponding bit of the **<Status Register>**’s Event register.

The corresponding query returns the current setting.

```
:STATus:<Status Register>:PTRansition?
```

Returns: <numeric>

:STATus:<Status Register>:NTRansition <numeric>

Sets the negative Transition filter. Setting a bit in the negative Transition Filter shall cause a 1 to 0 transition in the corresponding bit of the **<Status Register>**'s Condition register to cause a 1 to be written in the corresponding bit of the **<Status Register>**'s Event register.

The corresponding query returns the current setting.

:STATus:<Status Register>:NTRansition?

Returns: <numeric>

:STATus:<Status Register>:EVENt?

Returns: <numeric>

Returns the contents of the Event register associated with the **<Status Register>**. Reading this register clears its contents.

:STATus:<Status Register>:CONDition?

Returns: <numeric>

Returns the contents of the Condition register associated with the **<Status Register>**. Reading this register does not clear its contents.

:STATus:CHIStory <portid>

Clear the contents of all History registers for the given port.

Status Reporting Specific To SDH & Sonet

Status Reporting

The status reporting capability of the Instrument is provided by the STATus subsystem, its Status Registers and the Status Byte.

The following status registers are provided and conform to IEEE 488.2:

Status Register	Description
Standard Event	This register is accessed using the command *ESR?
QUEStionable	Defined by SCPI.
OPERation	Defined by SCPI.
INSTrument	Monitors general instrument status.
DATA	Monitors specific instrument status.
SONet:PORT or SDH:PORT	Line Level errors, alarms and events for one SONET/SDH signal.
SONet:PORT:SUMMary or SDH:PORT:SUMMary	Summary of the port status for every port allocated to a Test Session.
SONet:PATH or SDH:PATH	Status of the errors, alarms and events for one path within a SONET/SDH signal.
SONet:PATH:SUMMary or SDH:PATH:SUMMary	Summery of status of the errors, alarms and events for one path within a SONET/SDH signal.

General Status Register

The status registers conform to IEEE 488.2 and each comprises four registers. For the commands which access and control these registers, see “ STATus subsystem”.

Condition Monitors the defined Status conditions. There is no latching

Register of conditions in this register, it is updated in real time.

Transition Filter Determines whether positive or negative transitions (true or false) in the Condition Register set the Event Register.

Event Register Latches the transient states that occur in the Condition Register as specified by the Transition Filter.

Event Enable Register Masks the Event register, determining which of its bits set the summary bit in the Status Byte.

Status Byte

***STB?** or a serial port - Returns the value of the Status Byte in numeric form.

***SRE <numeric>** - Sets the Status Byte mask.

***SRE?** - Returns the current mask setting in numeric form.

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
OPER	RQS	ESR	MAV	QUES	-	-	-

DB0 - DB2	Not used, always read as 0.
DB3	QUES - QUEStionable Status register summary . Indicates that a bit has been set in the QUEStionable Status register.
DB4	MAV - Message Available. Remains set until err output messages are read from the Instrument.
DB5	ESR - Event Status register summary . Indicates that a bit has been set in the Event Status register.
DB6	RQS - Request Service. Set when en SRQ is generated for whatever reason. Cleared by SPOLL or *STB?
DB7	OPER - OPERation Status register summary . Indicates that a bit has been set in the OPERation Status register

Standard Event Status Register

***ESR?** - Returns the Standard Event Status Register value in numeric form.

***ESE** <numeric> - Sets the event enable register mask.

***ESE?** - Returns the current mask setting.

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
PWR	URQ	CME	EXE	DDE	QUE	RQC	OPC

DB0	OPC - OPERation Complete
DB1	RQC - Request Control.
DB2	QUE - Query Error
DB3	DDE - Device Dependent Error.
DB4	EXE - Execution Error.
DB5	CME - Command Error.
DB6	URQ - User Request.
DB7	PWR - Power On.

QUEStionable Status Register

Provides a summary of the DATA Status register.

DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	CMW	-	-	-	-	DATA	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
-	-	-	-	-	-	-	-

DB0-8	Not used, always read as 0
DB9	DATA - DATA Status register summary .
DB10-13	Not used, always read as 0.
DB14	CMW - Command Warning
DB15	Not used, always reads as 0

QUEStionable Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

STATus:QUEStionable:CONDition?

STATus:QUEStionable:ENABle <numeric>

STATus:QUEStionable:ENABle?

STATus:QUEStionable:EVENt?

STATus:QUEStionable:NTRansition <numeric>

STATus:QUEStionable:NTRansition?

STATus:QUEStionable:PTRansition <numeric>

STATus:QUEStionable:PTRansition?

OPERation Status Register

Provides a summary of the INSTrument Status register, and reports when a measurement is being made.

DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	-	INST	-	-	-	-	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
-	-	-	MEAS	-	-	-	-

DB0-3	Not used, always reads as 0
DB4	MEAS - Measuring. Currently making a measurement
DB5-12	Not used, always read as 0.
DB13	INST - INSTrument Status register summary .
DB14-15	Not used, always read as 0.

OPERational Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

STATus:OPERational:CONDition?

STATus:OPERational:ENABle <numeric>

STATus:OPERational:ENABle?

STATus:OPERational:EVENt?

STATus:OPERational:NTRansition <numeric>

STATus:OPERational:NTRansition?

STATus:OPERational:PTRansition <numeric>

STATus:OPERational:PTRansition?

INSTrument Status Register

Reports the instrument Status.

DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	-	-	-	-	-	-	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
	STP	STC	-	-	EOT	-	-

DB0	Reserved.
DB1	Reserved
DB2	EOT - End Of Test period.
DB3	Not used, always read as 0
DB4	Not used, always read as 0.
DB5	STC - Self Test complete.
DB6	STP - Last second period complete.
DB7	Reserved
DB8-15	Not used, always read as 0.

INSTrument Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

STATus:INSTrument:CONDition?

STATus:INSTrument:ENABle <numeric>

STATus:INSTrument:ENABle?

STATus:INSTrument:EVENt?

STATus:INSTrument:NTRansition <numeric>

STATus:INSTrument:NTRansition?

STATus:INSTrument:PTRansition <numeric>

STATus:INSTrument:PTRansition?

DATA Status Register

Summarizes the alarm Status registers shown. In addition provides a Power Failed alarm indication.

DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	-	-	-	-	-	-	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
-	-	-	-	-	SSUM	-	-

DB0- DB1	Reserved.
DB2	SSUM – The Summary message from the SONET/SDH Port Status Register.
DB3-DB14	Not used, always read as 0.
DB8-15	Not used and never will be.

DATA Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

STATus:DATA:CONDition?

STATus:DATA:ENABle <numeric>

STATus:DATA:ENABle?

STATus:DATA:EVENt?

STATus:DATA:NTRansition <numeric>

STATus:DATA:NTRansition?

STATus:DATA:PTRansition <numeric>

STATus:DATA:PTRansition?

SONet | SDH Path Status Register

This register set represents the status of the errors, alarms and events for one path within a SONET/SDH signal. To access these registers the <register> parameter should be replaced with **SONet:PATH** or **SDH:PATH**

DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	RES	PSL	-	BIT-ER	RDI-P	-	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
SDIR	PNTR	AIS-P	PDI-P	LOP	B3	REI-P	UNEQ

DB0	UNEQ	Indicates the specified path is unequipped
DB1	REI-P	Indicates Path REI on the specified path
DB2	B3	Indicates B3 Errors on the specified path
DB3	LOP	Indicates LOP on the specified path.
DB4	PDI-P	Indicates PDI-P on the specified path.
DB5	AIS-P	Indicates Path AIS on the specified path
DB6	PNTR	Indicates Pointer Adjust on the specified path.
DB7	SDIR	Indicates Service Disruption on the specified path
DB8	Not Used	
DB9	Not Used	
DB10	RDI-P	Indicates Path RDI on the specified path.
DB11	BIT	Indicates Bit Errors on the specified path.
DB12	Not Used	
DB13	PSL	Indicates Pattern Sync Loss on the specified path.
DB14	RES	Reserved for Summary of VT level events.
DB15	Not Used	And must never be used

SONet | SDH:PATH Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

:STATus:SONet:PATH:CONDition? <<channel>>, <portid>

:STATus:SONet:PATH:ENABle <numeric>, <<channel>>, <portid>

:STATus:SONet:PATH:ENABle? <channel>, <portid>

:STATus:SONet:PATH:EVENt? <channel>, <portid>

:STATus:SONet:PATH:NTRansition <numeric>, <channel>, <portid>

:STATus:SONet:PATH:NTRansition? <channel>, <portid>

:STATus:SONet:PATH:PTRansition <numeric>, <channel>, <portid>

:STATus:SONet:PATH:PTRansition? <channel>, <portid>

:STATus:SONet:PATH:HISTory? <channel>, <portid>

SONet | SDH:PATH:SUMMary Status Register

This register set represents the summary status of the errors, alarms and events for all paths on a given port . To access these registers the <register> parameter should be replaced with SONet:PATH:SUMMary or SDH:PATH:SUMMary

DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	RES	PSL	-	BIT-ER	RDI-P	-	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
SDIR	PNTR	AIS-P	PDI-P	LOP	B3	REI-P	UNEQ

DB0	UNEQ	Indicates the specified path is unequipped
DB1	REI-P	Indicates Path REI on the specified path
DB2	B3	Indicates B3 Errors on the specified path
DB3	LOP	Indicates LOP on the specified path.
DB4	PDI-P	Indicates PDI-P on the specified path.
DB5	AIS-P	Indicates Path AIS on the specified path
DB6	PNTR	Indicates Pointer Adjust on the specified path.
DB7	SDIR	Indicates Service Disruption on the specified path
DB8	Not Used	
DB9	Not Used	
DB10	RDI-P	Indicates Path RDI on the specified path.
DB11	BIT	Indicates Bit Errors on the specified path.
DB12	Not Used	
DB13	PSL	Indicates Pattern Sync Loss on the specified path.
DB14	RES	Reserved for Summary of VT level events.
DB15	Not Used	And must never be used

SONet | SDH:PATH:SUMMary Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

```
:STATus:SONet:PATH:SUMMary:CONDition? <portid>

:STATus:SONet:PATH:SUMMary:ENABle<numeric>, <portid>

:STATus:SONet:PATH:SUMMary:ENABle?      <portid>

:STATus:SONet:PATH:SUMMary:EVENt?       <portid>

:STATus:SONet:PATH:SUMMary:NTRansition <numeric>, <portid>

:STATus:SONet:PATH:SUMMary:NTRansition? <portid>

:STATus:SONet:PATH:SUMMary:PTRansition <numeric>,  <portid>

:STATus:SONet:PATH:SUMMary:PTRansition? <portid>

:STATus:SONet:PATH:SUMMary:HISTory?     <portid>
```

SONet | SDH Port Status Register

These registers represent the status of the Line Level errors, alarms and events for one SONET/SDH signal. To access these registers the <register> parameter should be replaced with **SONet:PORT** or **SDH:PORT**

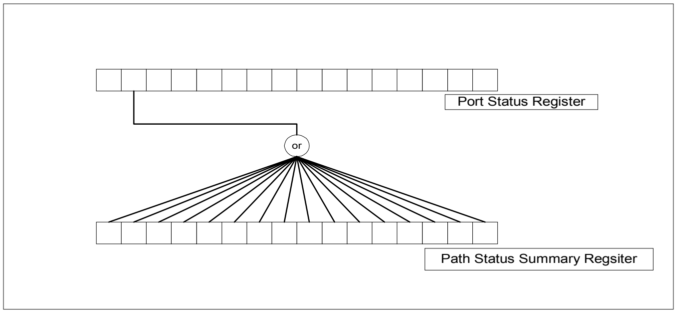
DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	SUMM-P	-	-	-	-	RDI-L	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
-	REI-L	B2	AIS-L	B1	OOF/SEF	LOF	LOS

DB0	LOS	Indicates LOS at the specified port
DB1	LOF	Indicates LOF at the specified port
DB2	SEF	Indicates SEF at the specified port
DB3	B1	Indicates B1 Errors on the specified port
DB4	AIS-L	Indicates Line AIS on the specified port
DB5	B2	Indicates B2 Errors on the specified port
DB6	REI-L	Indicates Line REI on the specified port.
DB7	Not Used	
DB8	Not Used	
DB9	RDI-L	Indicates Line RDI on the specified port.
DB10	Not Used	
DB11	Not Used	
DB12	Not Used	

DB13	Not Used	
DB14	SUMM-P	Channel Event Summary – Indicates an event or events are signalled in at least one of the received channels for the specified port.
DB15	Not Used	And must never be used

The Channel Event Summary bit (SUMM-P) is a summary of the Path Status Summary Register as shown below.



SONet | SDH:PORT Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

```
:STATus:SONet:PORT:CONDition? <portid>

:STATus:SONet:PORT:ENABle <numeric>,  <portid>

:STATus:SONet:PORT:ENABle? <portid>

:STATus:SONet:PORT:EVENt? <portid>

:STATus:SONet:PORT:NTRansition <numeric>,  <portid>

:STATus:SONet:PORT:NTRansition? <portid>

:STATus:SONet:PORT:PTRansition <numeric>,  <portid>

:STATus:SONet:PORT:PTRansition? <portid>

:STATus:SONet:PORT:HISTory? <portid>
```

SONet | SDH Port Status Summary Register.

These registers represent the summary of the port status for every port allocated to a Test Session. There will be only one set of these registers per Test Session. There are two registers each of which represents a summary of the status of all the ports within the session. To access these registers the <register> parameter should be replaced with **`SONet:PORT:SUMMary`** or **`SDH:PORT:SUMMary`**

DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8
-	SUMM-P	-	-	-	-	RDI-L	-

DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
-	REI-L	B2	AIS-L	B1	OOF/SEF	LOF	LOS

DB0	LOS	Indicates LOS at the specified port
DB1	LOF	Indicates LOF at the specified port
DB2	SEF	Indicates SEF at the specified port
DB3	B1	Indicates B1 Errors on the specified port
DB4	AIS-L	Indicates Line AIS on the specified port
DB5	B2	Indicates B2 Errors on the specified port
DB6	REI-L	Indicates Line REI on the specified port.
DB7	Not Used	
DB8	Not Used	
DB9	RDI-L	Indicates Line RDI on the specified port.
DB10	Not Used	
DB11	Not Used	
DB12	Not Used	
DB13	Not Used	

DB14	SUMM-P	Channel Event Summary – Indicates an event or events are signalled in at least one of the received channels for the specified port.
DB15	Not Used	And must never be used

SONet | SDH:PORT:SUMMary Status Register Commands

The following commands may be used to access the data stored in the above detailed register. The <numeric> parameter allows the user to enter a bit mask for the given register.

:STATus:SONet:PORT:SUMMary:CONDition?

:STATus:SONet:PORT:SUMMary:ENABle <numeric>

:STATus:SONet:PORT:SUMMary:ENABle?

:STATus:SONet:PORT:SUMMary:EVENt?

:STATus:SONet:PORT:SUMMary:NTRansition <numeric>,

:STATus:SONet:PORT:SUMMary:NTRansition?

:STATus:SONet:PORT:SUMMary:PTRansition <numeric>,

:STATus:SONet:PORT:SUMMary:PTRansition?

:STATus:SONet:PORT:SUMMary:HISTory?

SCPI Error Messages

The system-defined error/event numbers are chosen on an enumerated ("1 of N") basis. The SCPI defined error/event numbers and the error description portions of the ERRor query response are listed here. The first error/event described in each class (for example, -100, -200, -300, -400) is a "generic" error. In selecting the proper error/event number to report, more specific error/event codes are preferred, and the generic error/event is used only if the others are inappropriate.

No Error

This message indicates that the device has no errors.

No Error

The queue is completely empty. Every error/event in the queue has been read or the queue was purposely cleared by power-on, *CLS, etc.

Command Errors [-199, -100]

An < error/event number > in the range [-199, -100] indicates that an IEEE 488.2 syntax error has been detected by the instrument's parser. The occurrence of any error in this class should cause the command error bit (bit 5) in the event status register (IEEE 488.2, section 11.5.1) to be set. One of the following events has occurred:

An IEEE 488.2 system error has been detected by the parser. That is, a controller-to-device message was received which is in violation of the IEEE 488.2 standard. Possible violations include a data element which violates the device listening formats or whose type is unacceptable to the device.

An unrecognized header was received. Unrecognized headers include incorrect device-specific headers and incorrect or not implemented IEEE 488.2 common commands.

A Group Execute Trigger (GET) was entered into the input buffer inside of an IEEE 488.2 < PROGRAM MESSAGE >.

Events that generate command errors shall not generate execution errors, device-specific errors, or query errors.

-100 **Command error**

This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that a Command Error as defined in IEEE 488.2, 11.5.1.1.4 has occurred.

-101 **Invalid character**

A syntactic element contains a character which is invalid for that type; for example, a header containing an ampersand, SET UP&. This error might be used in place of errors -114, -121, -141, and perhaps some others.

-102 **Syntax error**

An unrecognized command or data type was encountered; for example, a string was received when the device does not accept strings.

-103 **Invalid separator**

The parser was expecting a separator and encountered an illegal character; for example, the semicolon was omitted after a program message unit, *ESE 1:OUTP1:TEL:RATE 140 Mb/s

-104 **Data type error**

The parser recognized a data element different than one allowed; for example, numeric or string data was expected but block data was encountered.

-105 **GET not allowed**

A Group Execute Trigger was received within a program message (see IEEE 488.2, 7.7).

-108 **Parameter not allowed**

More parameters were received than expected for the header; for example, the *RCL common command only accepts one parameter, so receiving *RCL 0,1 is not allowed.

-109 **Missing parameter**

Fewer parameters were received than required for the header; for example, the *ESE common command requires one parameter, so receiving *ESE is not allowed.

-110 **Command header error**

An error was detected in the header. This error message should be used when the device cannot detect the more specific errors described for errors -111 through -119.

-111 **Header separator error**

A character which is not a legal header separator was encountered while parsing the header; for example, no white space followed the header, thus *ESE1 is an error.

-112 **Program mnemonic too long**

The header contains more that twelve characters (see IEEE 488.2, 7.6.1.4.1).

-113 **Undefined header**

The header is syntactically correct, but it is undefined by this specific device; for example, *XYZ is not defined for any device.

-114 **Header suffix out of range**

Indicates that a non-header character has been encountered in what the parser expects is a header element.

-120 **Numeric data error**

This error, as well as errors -121 through -129, are generated when parsing a data element which appears to be numeric, including the non-decimal numeric types. This particular error message should be used if the device cannot detect a more specific error.

-121 **Invalid character in number**

An invalid character for the data type being parsed was encountered; for example, an alpha in a decimal numeric or a "9" in octal data.

-123 **Exponent too large**

The magnitude of the exponent was larger than 32000 (see IEEE 488.2, 7.7.2.4.1).

-124 **Too many digits**

The mantissa of a decimal numeric data element contained more than 255 digits excluding leading zeros (see IEEE 488.2, 7.7.2.4.1).

-128 **Numeric data not allowed**

A legal numeric data element was received, but the device does not accept one in this position for the header.

-130 **Suffix error**

This error, as well as errors -131 through -139, are generated when parsing a suffix. This particular error message should be used if the device cannot detect a more specific error.

-131 **Invalid suffix**

The suffix does not follow the syntax described in IEEE 488.2, 7.7.3.2, or the suffix is inappropriate for this device.

-134 **Suffix too long**

The suffix contained more than 12 characters (see IEEE 488.2, 7.7.3.4).

-138 **Suffix not allowed**

A suffix was encountered after a numeric element which does not allow suffixes.

-140 **Character data error**

This error, as well as errors -141 through -149, are generated when parsing a character data element. This particular error message should be used if the device cannot detect a more specific error.

-141 **Invalid character data**

Either the character data element contains an invalid character or the particular element received is not valid for the header.

-144 **Character data too long**

The character data element contains more than twelve characters (see IEEE 488.2, 7.7.1.4).

-148 **Character data not allowed**

A legal character data element was encountered where prohibited by the device.

-150 **String data error**

This error, as well as errors -151 through -159, are generated when parsing a string data element. This particular error message should be used if the device cannot detect a more specific error.

-151 **Invalid string data**

A string data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.5.2); for example, an END message was received before the terminal quote character.

-158 **String data not allowed**

A string data element was encountered but was not allowed by the device at this point in parsing.

-160 **Block data error**

This error, as well as errors -161 through -169, are generated when parsing a block data element. This particular error message should be used if the device cannot detect a more specific error.

-161 **Invalid block data**

A block data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.6.2); for example, an END message was received before the length was satisfied.

-168 **Block data not allowed**

A legal block data element was encountered but was not allowed by the device at this point in parsing.

-170 **Expression error**

This error, as well as errors -171 through -179, are generated when parsing an expression data element. This particular error message should be used if the device cannot detect a more specific error.

-171 **Invalid expression**

The expression data element was invalid (see IEEE 488.2, 7.7.7.2); for example, unmatched parentheses or an illegal character.

-178 **Expression data not allowed**

A legal expression data was encountered but was not allowed by the device at this point in parsing.

-180 **Macro error**

This error, as well as errors -181 through -189, are generated when defining a macro or executing a macro. This particular error message should be used if the device cannot detect a more specific error.

-181 **Invalid outside macro definition**

Indicates that a macro parameter placeholder ($<number) was encountered outside of a macro definition.

-183 **Invalid inside macro definition**

Indicates that the program message unit sequence, sent with a *DDT or *DMC command, is syntactically invalid (see 10.7.6.3).

-184 **Macro parameter error**

Indicates that a command inside the macro definition had the wrong number or type of parameters.

Execution Errors [-299, -200]

An < error/event number > in the range [-299, -200 indicates that an error has been detected by the instrument's execution control block. The occurrence of any error in this class should cause the execution error bit (bit 4) in the event status register (IEEE 488.2, section 11.5.1) to be set. One of the following events has occurred:

A < PROGRAM DATA > element following a header was evaluated by the device as outside of its legal input range or is otherwise inconsistent with the device

A valid program message could not be properly executed due to some device condition.

Execution errors shall be reported by the device after rounding and expression evaluation operations have taken place. Rounding a numeric data element, for example, shall not be reported as an execution error. Events that generate execution errors shall not generate Command Errors, device-specific errors, or Query Errors.

-200 **Execution error**

This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that an Execution Error as defined in IEEE 488.2, 11.5.1.1.5 has occurred.

-201 **Invalid while in local**

Indicates that a command is not executable while the device is in local due to a hard local control (see IEEE 488.2, 5.6.1.5); for example, a device with a rotary switch receives a message which would change the switches state, but the device is in local so the message can not be executed.

-202 **Settings lost due to rtl**

Indicates that a setting associated with a hard local control (see IEEE 488.2, 5.6.1.5) was lost when the device changed to LOCS from REMS or to LWLS from RWLS.

-210 **Trigger error**

-211 **Trigger ignored**

Indicates that a GET, *TRG, or triggering signal was received and recognized by the device but was ignored because of device timing considerations; for example, the device was not ready to respond. Note: a DT0 device always ignores GET and treats *TRG as a Command Error.

-212 **Arm ignored**

Indicates that an arming signal was received and recognized by the device but was ignored.

-213 **Init ignored**

Indicates that a request for a measurement initiation was ignored as another measurement was already in progress.

-214 **Trigger deadlock**

Indicates that the trigger source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.

-215 **Arm deadlock**

Indicates that the arm source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.

-220 **Parameter error**

Indicates that a program data element related error occurred. This error message should be used when the device cannot detect the more specific errors described for errors -221 through -229.

-221 **Setting conflict**

Indicates that a legal program data element was parsed but could not be executed due to the current device state (see IEEE 488.2, 6.4.5.3 and 11.5.1.1.5.)

-222 **Data out of range**

Indicates that a legal program data element was parsed but could not be executed because the interpreted value was outside the legal range as defined by the device (see IEEE 488.2, 11.5.1.1.5.)

-223 **Too much data**

Indicates that a legal program data element of block, expression, or string type was received that contained more data than the device could handle due to memory or related device-specific requirements.

-224 **Illegal parameter value**

Used where exact value, from a list of possibles, was expected.

-230 **Data corrupt or stale**

Possibly invalid data; new reading started but not completed since last access.

-231 **Data questionable**

Indicates that measurement accuracy is suspect.

-240 **Hardware error**

Indicates that a legal program command or query could not be executed because of a hardware problem in the device. Definition of what constitutes a hardware problem is completely device-specific. This error message should be used when the device cannot detect the more specific errors described for errors -241 through -249.

-241 **Hardware missing**

Indicates that a legal program command or query could not be executed because of missing device hardware; for example, an option was not installed. Definition of what constitutes missing hardware is completely device-specific.

-250 **Mass storage error**

Indicates that a mass storage error occurred. This error message should be used when the device cannot detect the more specific errors described for errors -251 through -259.

-251 **Missing mass storage**

Indicates that a legal program command or query could not be executed because of missing mass storage; for example, an option that was not installed. Definition of what constitutes missing mass storage is device-specific.

-252 **Missing media**

Indicates that a legal program command or query could not be executed because of a missing media; for example, no disk. The definition of what constitutes missing media is device-specific.

-253 **Corrupt media**

Indicates that a legal program command or query could not be executed because of corrupt media; for example, bad disk or wrong format. The definition of what constitutes corrupt media is device-specific.

-254 **Media full**

Indicates that a legal program command or query could not be executed because the media was full; for example, there is no room on the disk. The definition of what constitutes a full media is device-specific.

-255 **Directory full**

Indicates that a legal program command or query could not be executed because the media directory was full. The definition of what constitutes a full media directory is device-specific.

-256 **File name not found**

Indicates that a legal program command or query could not be executed because the file name on the device media was not found; for example, an attempt was made to read or copy a nonexistent file. The definition of what constitutes a file not being found is devicespecific.

-257 **File name error**

Indicates that a legal program command or query could not be executed because the file name on the device media was in error; for example, an attempt was made to copy to a duplicate file name. The definition of what constitutes a file name error is device-specific.

-258 **Media protected**

Indicates that a legal program command or query could not be executed because the media was protected; for example, the writeprotect tab on a disk was present. The definition of what constitutes protected media is device-specific.

-260 **Expression error**

Indicates that an expression program data element related error occurred. This error message should be used when the device cannot detect the more specific errors described for errors -261 through -269.

-261 **Math error in expression**

Indicates that a syntactically legal expression program data element could not be executed due to a math error; for example, a divide-byzero was attempted. The definition of math error is device-specific.

-270 **Macro error**

Indicates that a macro-related execution error occurred. This error massage should be used when the device cannot detect the more specific errors described for errors -271 through -279.

-271 **Macro syntax error**

Indicates that a syntactically legal macro program data sequence, according to IEEE 488.2, 10.7.2, could not be executed due to a syntax error within the macro definition (see IEEE 488.2, 10.7.6.3.)

-272 **Macro execution error**

Indicates that a syntactically legal macro program data sequence could not be executed due to some error in the macro definition (see IEEE 488.2, 10.7.6.3.)

-273 **Illegal macro label**

Indicates that the macro label defined in the *DMC command was a legal string syntax but could not be accepted by the device (see IEEE 488.2, 10.7.3 and 10.7.6.2); for example, the label was too long, the same as a common command header, or contained invalid header syntax.

-274 **Macro parameter error**

Indicates that the macro definition improperly used a macro parameter placeholder (see IEEE 488.2, 10.7.3).

-275 **Macro definition too long**

Indicates that a syntactically legal macro program data sequence could not be executed because the string or block contents were too long for the device to handle (see IEEE 488.2, 10.7.6.1).

-276 **Macro recursion error**

Indicates that a syntactically legal macro program data sequence could not be executed because the device found it to be recursive (see IEEE 488.2, 10.7.6.6).

-277 **Macro redefinition not allowed**

Indicates that a syntactically legal macro label in the *DMC command could not be executed because the macro label was already defined (see IEEE 488.2, 10.7.6.4).

-278 **Macro header not found**

Indicates that a syntactically legal macro label in the *GMC? query could not be executed because the header was not previously defined.

-280 **Program error**

Indicates that a downloaded program-related execution error occurred. This error message should be used when the device cannot detect the more specific errors described for errors -281 through -289.

A downloaded program is used to add algorithmic capability to a device. The syntax used in the program and the mechanism for downloading a program is device-specific.

-281 **Cannot create program**

Indicates that an attempt to create a program was unsuccessful. A reason for the failure might include not enough memory.

-282 **Illegal program name**

The name used to reference a program was invalid; for example, redefining an existing program, deleting a nonexistent program, or in general, referencing a nonexistent program.

-283 **Illegal variable name**

An attempt was made to reference a nonexistent variable in a program.

-284 **Program currently running**

Certain operations dealing with programs may be illegal while the program is running; for example, deleting a running program might not be possible.

-285 **Program syntax error**

Indicates that a syntax error appears in a downloaded program. The syntax used when parsing the downloaded program is device-specific.

-286 **Program runtime error**

Query Errors [-399, -300]

An < error/event number > in the range [-399, -300] indicates that the instrument has detected an error which is not a command error, a query error, or an execution error; some device operations did not properly complete, possibly due to an abnormal hardware or firmware condition. These codes are also used for self-test response errors. The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the event status register (IEEE 488.2, section 11.5.1) to be set. The meaning of positive error codes is device-dependent and may be enumerated or bit mapped; the <error message> string for positive error codes is not defined by SCPI and available to the device engineer. Note that the string is not optional; if the designer does not wish to implement a string for a particular error, the null string should be sent (for example 42," "). The occurrence of any error in this class should cause the device-specific error bit (bit 3) in the event status register (IEEE 488.2, section 11.5.1) to be set. Events that generate device-specific errors shall not generate command errors, execution errors, or query errors; see the other error definitions in this section.

-300 **Device-specific error**

This is the generic device-dependent error for devices that cannot detect more specific errors. This code indicates only that a Device-Dependent Error as defined in IEEE 488.2, 11.5.1.1.6 has occurred.

-310 **System error**

Indicates that some error, termed "system error" by the device has occurred. This code is device-dependent.

-311 **Memory error**

Indicates that an error was detected in the device's memory. The scope of this error is device-dependent.

-312 **PUD memory lost**

Indicates that the protected user data saved by the *PUD command has been lost.

-313 **Calibration memory lost**

Indicates that non-volatile calibration data used by the *CAL? command has been lost.

-314 **Save/Recall memory lost**

Indicates that the non-volatile data saved by the *SAV? command has been lost.

-315 **Configuration memory lost**

Indicates that the non-volatile data saved by the device has been lost. The meaning of this error is device-specific.

-330 **Self-test failed**

-350 **Queue overflow**

A specific code entered into the queue in lieu of the code that caused the error. This code indicates that there is no room in the queue and an error occurred but was not recorded.

-360 **Communication error**

This is the generic communication error for devices that cannot detect the more specific errors described for errors -361 through -363.

-361 **Parity error in program message**

Parity bit not correct when data received for example, on a serial port (for example, a baud rate mismatch).

-362 **Framing error in program message**

A stop bit was not detected when data was received for example, on a serial port.

-363 **Input buffer overrun**

Software or hardware input buffer on serial port overflows with data caused by improper or non-existent pacing.

Query Errors [-499, -400]

An < error/event number > in the range [-499, -400] indicates that the output queue control of the instrument has detected a problem with the message exchange protocol described in IEEE 488.2, topic 6. The occurrence of any error in this class should cause the query error bit (bit 2) in the event status register (IEEE 488.2, section 11.5.1) to be set. These errors correspond to message exchange protocol errors described in IEEE 488.2, section 6.5. One of the following is true:

An attempt is being made to read data from the output queue when no output is either present or pending;

Data in the output queue has been lost.

Events that generate query errors shall not generate command errors, execution errors, or device-specific errors; see the other error definitions in this section.

-400 **Query error**

This is the general query error for devices that cannot detect more specific errors. This code indicates only that a Query Error as defined in IEEE 488.2, 11.5.1.1.7 and 6.3 has occurred.

-410 **Query INTERRUPTED**

Indicates that a condition causing an INTERRUPTED Query error occurred (see IEEE 488.2, 6.3.2.3); for example, a query followed by DAB or GET before a response was completely sent.

-420 **Query UNTERMINATED**

Indicates that a condition causing an UNTERMINATED Query error occurred (see IEEE 488.2, 6.3.2.2); for example, the device was addressed to talk and an incomplete program message was received.

-430 **Query DEADLOCKED**

Indicates that a condition causing a DEADLOCKED Query error occurred (see IEEE 488.2, 6.3.1.7); for example, both input buffer and output buffer are full and the device cannot continue.

-440 **Query UNTERMINATED after indefinite response**

Indicates that a query was received in the same program message after a query requesting an indefinite response was executed (see IEEE 488.2, 6.5.7.5.7.)

OmniBerXM SCPI Specific Errors [-2065, -2026]

An < error/event number > in the range [-2065, -3000] indicates that the output queue control of the instrument has detected a problem with fuctionality specific and unique to the OmniberXm patform such as problems with the session or ports. The following error messages may be returned as a result of a problem in the underlying system:

-2026 **In Parameter Count Invalid**

-2029 **Invoke Unsuccesful**

-2030 **Method Exception**

-2033 **Parameter Name Null**

-2035 **No Session Opened**

-2044 **Param Count Invalid**

-2046 **Type Mismatch**

-2061 **Advise Not Succesful**

-2063 **IW Default Port Group Null**

-2064 **IW No Port Available**

-2062 **Scpi Command In Progress**

-2065 **Stat Port Not Selected**

OmniBerXM SCPI Specific Errors [-4000, -5000]

Errors within this range indicate an error hs been returned from the underlying System API. For further details on these errors, please consult the System API documentation.

SMANAGER:SESSION:OPEN <STRING> <DISCRETE> 21

SMANAGER:SESSION:OPEN? 21

SMANAGER:SESSION:CLOSE <NUMERIC> 21

SMANAGER:SESSION:LIST:OPEN? 21

SMANAGER:SESSION:LIST:TYPES? 21

SMANAGER:SESSION:TYPE? <NUMERIC> 22

SMANAGER:SESSION:CONTEXT? <NUMERIC> 22

SMANAGER:SESSION:SOCKET? <NUMERIC> 22

SMANAGER:SESSION:LABEL <NUMERIC> <STRING> 22

SMANAGER:SESSION:LABEL? <NUMERIC> 23

SMANAGER:SESSION:PID? <NUMERIC> 23

SMANAGER:SESSION:GUI:CONNECTIONS? <NUMERIC> <STRING> 23

SMANAGER:SESSION:GUI:MCONNECT <STRING> <NUMERIC> 24

SMANAGER:SESSION:GUI:MCONNECT? <STRING> 24

SMANAGER:MODULE:UNLOCK <STRING> 25

SMANAGER:MODULE:REBOOT:MODULE <STRING> 25

SMANAGER:MODULE:REBOOT:ALL 25

SMANAGER:MODULE:UPDATE:ALL 25

SMANAGER:MODULE:UPDATE:AUTOUPDATE <BOOLEAN> 25

SMANAGER:MODULE:UPDATE:AUTOUPDATE? 26

SMANAGER:MODULE:USMODULE <STRING>..26

SMANAGER:MODULE:LIST? 26

SMANAGER:MODULE:LALL? 27

SMANAGER:MODULE:SERIAL? <MODULENUMBER> 27

SMANAGER:MODULE:NUMBER? <STRING>..27

SMANAGER:MODULE:LIST:SAVE...27

SMANAGER:MODULE:LIST:SAVE?..28

SMANAGER:MODULE:LIST:NEW?...28

SMANAGER:MODULE:LIST:MISSING?...28

SMANAGER:MODULE:DESCRIPTION? <STRING>..28

SMANAGER:MODULE:PTYPE? <STRING> <NUMERIC>......................................28

SMANAGER:MODULE:NCPCOUNT? <STRING> 29

SMANAGER:MODULE:SYNCHRONISED? <STRING> 29

SMANAGER:MODULE:DUMMY? <STRING>...29

SMANAGER:MODULE:CHASSIS:NUMBER? <STRING> 29

SMANAGER:MODULE:CHASSIS:SLOT? <STRING> 30

SMANAGER:MODULE:ANNOTATION <STRING> <STRING>..............................30

SMANAGER:MODULE:ANNOTATION? <STRING>..30

SMANAGER:MODULE:CMASTER? <STRING>...30

SMANAGER:MODULE:IPADDRESS? <STRING> <NUMERIC> 31

SMANAGER:MODULE:IPADDRESS:PRIMARY? <STRING>..................................31

SMANAGER:MODULE:IPADDRESS:HOST? 31

SMANAGER:MODULE:STATE? <STRING> 31

SMANAGER:MODULE:LOCK? <STRING> 32

SMANAGER:MODULE:SDOWN? <STRING> 32

SMANAGER:MODULE:SDOWN <STRING> 32

SMANAGER:MODULE:LED:FLASH <STRING> 32

SMANAGER:MODULE:LED:IP 33

TSESSION:OPEN <STRING> <DISCRETE> 34

TSESSION:OPEN? 34

TSESSION:CLOSE 34

TSESSION:FSCLOSE 34

TSESSION:NPORTS? 34

TSESSION:HANDLE? 35

TSESSION:TYPE? 35

TSESSION:MODE? 35

TSESSION:CONTEXT? 35

TSESSION:SAVE <STRING> 35

TSESSION:RESTORE <STRING> 36

TSESSION:INTERFACE:SAVEABLE? 36

TSESSION:RESET 36

TSESSION:RESET:INTERFACES <LIST OF STRINGS> 36

TSESSION:DISCONNECT 37

TSESSION:DISCONNECT? 37

TSESSION:PORT: COMMANDS.......... 38

TSESSION:PORT:ADD <PORTID>.......... 38

TSESSION:PORT:ADD:LIST <LIST> 38

TSESSION:PORT:REMOVE <PORTID> 38

TSESSION:PORT:REMOVE:LIST <LIST>.......... 39

TSESSION:PORT:LIST?.......... 39

TSESSION:PORT:DETAILS? <PORTID>.......... 39

TSESSION:PORT:DUMMY? <PORTID>.......... 39

TSESSION:PORT:MODULE:ADD <STRING> 40

TSESSION:PORT:MODULE:REMOVE <MODULENUMBER>.......... 40

TSESSION:PORT:MODULE:PLIST? <MODULENUMBER> 40

TSESSION:PORT:COMMENT <STRING> <PORTID> 40

TSESSION:PORT:COMMENT? <PORTID> 41

SYSTEM:MODULE:LIST?.......... 42

SYSTEM:MODULE:LAST? 42

SYSTEM:MODULE:DESCRIPTION <MODULENUMBER>.......... 42

SYSTEM:MODULE:SYNCHRONISED? <MODULENUMBER> 42

SYSTEM:CHASSIS:NUMBER? <MODULENUMBER> 43

SYSTEM:CHASSIS:SLOT? <MODULENUMBER>.......... 43

SYSTEM:TYPE? <PORTID>..43

SYSTEM:MODULE:TYPES? <MODULENUMBER>...43

SYSTEM:MODULE:TYPE <MODULENUMBER> <DISCRETE> 44

SYSTEM:MODULE:LIMIT? 44

SYSTEM:MODULE:SUPPORTED? <MODULENUMBER> 44

SYSTEM:MODULE:STATE? <MODULENUMBER> 44

SYSTEM:MODULE:LOCK? <MODULENUMBER> 45

SYSTEM:MODULE:LOCK:UPSTREAM? <NUMERIC> 45

SYSTEM:MODULE:LOCK:DOWNSTREAM? <NUMERIC> 45

SYSTEM:MODULE:LOCK:LIST?...46

SYSTEM:MODULE:REQUIRED? <STRING> 46

SYSTEM:MODULE:UNAVAILABLE? <STRING>..46

:OUTPUT:TELECOM:LASER <BOOLEAN> <PORTID> 48

:OUTPUT:TELECOM:LASER? <PORTID>...48

:OUTPUT:TELECOM:LASER:ALL <BOOLEAN> 48

:SOURCE:DATA:TELECOM:MODE <DISCRETE> <PORTID>...............................49

:SOURCE:DATA:TELECOM:MODE? <PORTID>..49

:SOURCE:DATA:TELECOM:THRU <BOOLEAN> <PORTID>................................49

:SOURCE:DATA:TELECOM:THRU? <PORTID>...49

:SOURCE:CLOCK:SOURCE <DISCRETE> <PORTID> 50

:SOURCE:CLOCK:SOURCE? <PORTID> 50

:SOURCE:DATA:TELECOM:PATTERN:TYPE <DISCRETE> <CHANNEL> <PORTID> 51

:SOURCE:DATA:TELECOM:PATTERN:TYPE? <CHANNEL> <PORTID> 51

:SOURCE:DATA:TELECOM:PATTERN:TYPE:WORD <DISCRETE> <PORTID>.. 51

:SOURCE:DATA:TELECOM:PATTERN:TYPE:WORD? <PORTID> 51

:SOURCE:DATA:TELECOM:PATTERN:TYPE:WORD:USER <NUMERIC> <PORTID> 52

:SOURCE:DATA:TELECOM:PATTERN:TYPE:WORD:USER? <PORTID> 52

:SOURCE:DATA:TELECOM:ALARM <BOOLEAN> <PORTID> 53

:SOURCE:DATA:TELECOM:ALARM? <PORTID> 53

SOURCE:DATA:TELECOM:ALARM:MODE? <PORTID> 54

TIMED ALARM PERIODS 54

:SOURCE:DATA:TELECOM:ALARM:STRESS:TIMED <NUMERIC> <NUMERIC> <NUMERIC> <PORTID> 54

:SOURCE:DATA:TELECOM:ALARM:STRESS:TIMED? <PORTID> 54

:SOURCE:DATA:TELECOM:ALARM:STRESS:TIMED:START <PORTID> 55

:SOURCE:DATA:TELECOM:ALARM:STRESS:TIMED:STOP <PORTID> 55

:SOURCE:DATA:TELECOM:ALARM:STRESS:TIMED:RUNNING? <PORTID>.... 55

:SOURCE:DATA:TELECOM:ALARM:STRESS:PULSE <NUMERIC> <PORTID>.. 55

:SOURCE:DATA:TELECOM:ALARM:STRESS:PULSE? <PORTID>........ 55

:SOURCE:DATA:TELECOM:ALARM:STRESS:PULSE:START <PORTID>........ 56

:SOURCE:DATA:TELECOM:ERROR:SINGLE <PORTID>........ 57

:SOURCE:DATA:TELECOM:ERROR:SINGLE:ALL.. 57

SOURCE:DATA:TELECOM:ERROR:MODE? <PORTID>.. 57

:SOURCE:DATA:TELECOM:ERROR:STRESS:TIMED <NUMERIC> <NUMERIC> <NUMERIC> <PORTID> 57

:SOURCE:DATA:TELECOM:ERROR:STRESS:TIMED? <PORTID> 58

:SOURCE:DATA:TELECOM:ERROR:STRESS:TIMED:START <PORTID> 58

:SOURCE:DATA:TELECOM:ERROR:STRESS:TIMED:STOP <PORTID> 58

:SOURCE:DATA:TELECOM:ERROR:STRESS:TIMED:RUNNING? <PORTID>..... 58

:SENSE:DATA:TELECOM:MODE <DISCRETE> <PORTID>.................................... 60

:SENSE:DATA:TELECOM:MODE? <PORTID> 60

:SENSE:DATA:TELECOM:PATTERN:TYPE <DISCRETE> <CHANNEL> <PORTID>... 61

:SENSE:DATA:TELECOM:PATTERN:TYPE? <CHANNEL> <PORTID>................. 61

:SENSE:DATA:TELECOM:PATTERN:TYPE:WORD <DISCRETE> <PORTID>..... 61

:SENSE:DATA:TELECOM:PATTERN:TYPE:WORD? <PORTID>............................ 61

:SENSE:DATA:TELECOM:PATTERN:TYPE:WORD:USER <NUMERIC> <PORTID>... 62

:SENSE:DATA:TELECOM:PATTERN:TYPE:WORD:USER? <PORTID>................. 62

:SENSE:DATA:TELECOM:TEST <BOOLEAN>... 62

:SENSE:DATA:TELECOM:TEST? 62

SENSE:DATA:TELECOM:TEST:PORT:SELECT <LIST> 63

SENSE:DATA:TELECOM:TEST:PORT:SELECT? 63

SENSE:DATA:TELECOM:TEST:PORT:ANALYSIS:MODE <PORTID> <DISCRETE> 63

SENSE:DATA:TELECOM:TEST:PORT:ANALYSIS:MODE? <PORTID> 63

:SENSE:DATA:TELECOM:TEST:TYPE <DISCRETE> 64

:SENSE:DATA:TELECOM:TEST:TYPE? 64

:SENSE:DATA:TELECOM:TEST:PERIOD <NUMERIC>, <NUMERIC>, <NUMERIC>, <NUMERIC> 64

:SENSE:DATA:TELECOM:TEST:PERIOD? 64

:SENSE:DATA:TELECOM:TEST:START? 64

:SENSE:DATA:TELECOM:TEST:ELAPSED? 65

:SENSE:DATA:TELECOM:SDGT:TIME <NUMERIC> <PORTID> 65

:SENSE:DATA:TELECOM:SDGT:TIME? <PORTID> 65

:SENSE:DATA:TELECOM:SDGT:DEFAULT <PORTID> 65

:SENSE:DATA? <STRING> <PORTID> 66

:SENSE:DATA? <STRING> <PORTID> 67

:SENSE:DATA? <STRING> <PORTID> 67

:SYSTEM:PRESET 68

:SYSTEM:SERIAL? 68

:SYSTEM:VERSION? 68

:SYSTEM:ERROR? 68

SYSTEM:LOGGING:ERROR <BOOLEAN> 69

SYSTEM:LOGGING:ERROR:FILE <STRING> 69

SYSTEM:LOGGING:ERROR:FILE? <STRING> 70

SYSTEM:LOGGING:ERROR:PORTS <STRING> 70

SYSTEM:LOGGING:ERROR:PORTS? 70

SYSTEM:LOGGING:ERROR:AVAILABLE? 70

SYSTEM:LOGGING:ERROR:SELECT <STRING> 71

SYSTEM:LOGGING:ERROR:SELECT? 71

SYSTEM:LOGGING:ALARM <BOOLEAN> 72

SYSTEM:LOGGING:ALARM? 72

SYSTEM:LOGGING:ALARM:FILE <STRING> 72

SYSTEM:LOGGING:ALARM:FILE? 73

SYSTEM:LOGGING:ALARM:PORTS <STRING> 73

SYSTEM:LOGGING:ALARM:PORTS? 73

SYSTEM:LOGGING:ALARM:AVAILABLE? 73

SYSTEM:LOGGING:ALARM:SELECT <STRING> 73

SYSTEM:LOGGING:ALARM:SELECT? 74

SYSTEM:LOGGING:STATISTICS <BOOLEAN> 75

SYSTEM:LOGGING:STATISTICS? 75

SYSTEM:LOGGING:STATISTICS:FILE <STRING> 75

SYSTEM:LOGGING:STATISTICS:FILE? 76

SYSTEM:LOGGING:STATISTICS:INTERVAL <NUMERIC> 76

SYSTEM:LOGGING:STATISTICS:INTERVAL? 76

SYSTEM:LOGGING:STATISTICS:PORTS <STRING>..76

SYSTEM:LOGGING:STATISTICS:PORTS?..76

SYSTEM:LOGGING:STATISTICS:AVAILABLE? 77

SYSTEM:LOGGING:STATISTICS:SELECT <STRING>.......................................77

SYSTEM:LOGGING:STATISTICS:SELECT?..78

*CLS...79

*ESE <NUMERIC>...79

*ESE?..79

*ESR?..79

*IDN?..79

*OPC...80

*OPC?..80

*OPT?..80

*RST...80

*SRE <NUMERIC>...80

*SRE?..81

*STB?..81

:SOURCE:DATA:TELECOM:SDH:RATE <DISCRETE> <PORTID> 83

:SOURCE:DATA:TELECOM:SDH:RATE? <PORTID>...83

:SOURCE:DATA:TELECOM:SDH:SCRAMBLING <BOOLEAN> <PORTID>.........83

:SOURCE:DATA:TELECOM:SDH:SCRAMBLING? <PORTID> 83

:SOURCE:DATA:TELECOM:SDH:CHANNELS <STRING> <PORTID> 84

:SOURCE:DATA:TELECOM:SDH:CHANNELS? <PORTID> 84

:SOURCE:DATA:TELECOM:SDH:CHANNELS:EQUIPPED <LIST> <PORTID> 85

:SOURCE:DATA:TELECOM:SDH:CHANNELS:EQUIPPED? <PORTID> 85

:SOURCE:DATA:TELECOM:SDH:CHANNELS:UNEQUIPPED <LIST> <PORTID> 85

:SOURCE:DATA:TELECOM:SDH:CHANNELS:INVERT <PORTID> 85

:SOURCE:DATA:TELECOM:SDH:CHANNELS:ALL <PORTID> 86

:SOURCE:DATA:TELECOM:SDH:CHANNELS:AT? <NUMERIC> <PORTID> 86

:SOURCE:DATA:TELECOM:SDH:CHANNELS:SPOSITION? <NUMERIC> <PORTID> 86

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:DEFAULT <PORTID> 87

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:DATA <NUMERIC>, <NUMERIC>, <DISCRETE>, <NUMERIC> <PORTID> 87

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:DATA? <NUMERIC>, <NUMERIC> <DISCRETE> <PORTID> 88

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:J0:PATTERN <DISCRETE> <PORTID> 88

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:J0:PATTERN? <PORTID> 89

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:J0:PATTERN:B16CRC <STRING> <PORTID> 89

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:J0:PATTERN:B16CRC? <PORTID> 89

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:J0:PATTERN:B64 <STRING> <PORTID> 90

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:J0:PATTERN:B64? <PORTID> 90

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:DEFAULT <CHANNEL> <PORTID> 90

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:DATA <DISCRETE>,<NUMERIC> <CHANNEL> <PORTID> 90

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:DATA? <DISCRETE> <CHANNEL> <PORTID> 91

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:J1:PATTERN <DISCRETE> <CHANNEL> <PORTID> 91

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:J1:PATTERN? <CHANNEL> <PORTID> 92

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:J1:PATTERN:B16CRC <STRING> <CHANNEL> <PORTID> 92

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:J1:PATTERN:B16CRC? <CHANNEL> <PORTID> 92

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:J1:PATTERN:B64 <STRING> <CHANNEL> <PORTID> 93

:SOURCE:DATA:TELECOM:SDH:POVERHEAD:J1:PATTERN:B64? <CHANNEL> <PORTID> 93

:SOURCE:DATA:TELECOM:SDH:ERROR:CHANNELS:SELECTED <LIST> <PORTID> 94

:SOURCE:DATA:TELECOM:SDH:ERROR:CHANNELS:SELECTED ? <PORTID> 94

:SOURCE:DATA:TELECOM:SDH:ERROR:CHANNELS:UNSELECTED <LIST> <PORTID> 94

:SOURCE:DATA:TELECOM:SDH:ERROR:CHANNELS:INVERT <PORTID> 95

:SOURCE:DATA:TELECOM:SDH:ERROR:CHANNELS:ALL <PORTID> 95

:SOURCE:DATA:TELECOM:SDH:ERROR:GROUP <DISCRETE> <PORTID>.......95

:SOURCE:DATA:TELECOM:SDH:ERROR:GROUP? <PORTID>..............................95

:SOURCE:DATA:TELECOM:SDH:ERROR:SECTION <DISCRETE> <PORTID>...95

:SOURCE:DATA:TELECOM:SDH:ERROR:SECTION? <PORTID> 96

:SOURCE:DATA:TELECOM:SDH:ERROR:SECTION:MSREI <DISCRETE>
<PORTID>...96

:SOURCE:DATA:TELECOM:SDH:ERROR:SECTION:MSREI? <PORTID>.............96

:SOURCE:DATA:TELECOM:SDH:ERROR:PATH <DISCRETE> <PORTID>.........96

:SOURCE:DATA:TELECOM:SDH:ERROR:PATH? <PORTID>................................96

:SOURCE:DATA:TELECOM:SDH:ERROR:PAYLOAD <DISCRETE> <PORTID> 97

:SOURCE:DATA:TELECOM:SDH:ERROR:PAYLOAD? <PORTID>........................97

:SOURCE:DATA:TELECOM:ERROR:SINGLE <PORTID>.......................................97

:SOURCE:DATA:TELECOM:SDH:ERROR:RATE:USER <DISCRETE>
<NUMERIC> <DISCRETE> <PORTID>..97

:SOURCE:DATA:TELECOM:SDH:ERROR:RATE:USER? <DISCRETE> <PORTID>
....98

:SOURCE:DATA:TELECOM:SDH:ERROR:RATE:USER:ACTION <BOOLEAN>
<PORTID>...99

:SOURCE:DATA:TELECOM:SDH:ERROR:RATE:USER:ACTION:ALL
<BOOLEAN> 99

:SOURCE:DATA:TELECOM:SDH:ALARM:GROUP <DISCRETE> <PORTID>....100

:SOURCE:DATA:TELECOM:SDH:ALARM:GROUP? <PORTID> 100

:SOURCE:DATA:TELECOM:SDH:ALARM:SECTION <DISCRETE> <PORTID>.100

:SOURCE:DATA:TELECOM:SDH:ALARM:SECTION? <PORTID> 101

:SOURCE:DATA:TELECOM:SDH:ALARM:OOF <PORTID>.................................. 101

:SOURCE:DATA:TELECOM:SDH:ALARM:PATH <DISCRETE> <PORTID>....... 101

:SOURCE:DATA:TELECOM:SDH:ALARM:PATH? <PORTID>.............................. 101

:SOURCE:DATA:TELECOM:SDH:ALARM:PATH:ERDIP:MODE <BOOLEAN> <PORTID>... 101

:SOURCE:DATA:TELECOM:SDH:ALARM:PATH:ERDIP:MODE? <PORTID> 102

:SOURCE:DATA:TELECOM:SDH:ALARM:PATH:VALUE <DISCRETE> <NUMERIC> <PORTID> 102

:SOURCE:DATA:TELECOM:SDH:ALARM:PATH:VALUE? <DISCRETE> <PORTID>... 102

:SOURCE:DATA:TELECOM:SDH:ALARM:PATH:VALUE:DEFAULT <DISCRETE> <PORTID>... 102

:SOURCE:DATA:TELECOM:SDH:POINTER:VALUE <NUMERIC> <BOOLEAN> <PORTID>... 103

:SOURCE:DATA:TELECOM:SDH:POINTER:VALUE? <PORTID>........................ 103

:SOURCE:DATA:TELECOM:SDH:POINTER:VALUE:INCREMENT <PORTID>. 103

:SOURCE:DATA:TELECOM:SDH:POINTER:VALUE:DECREMENT <PORTID>103

:SOURCE:CLOCK:SDH:FOFFSET:OFFSET <NUMERIC><PORTID>.................... 104

:SOURCE:CLOCK:SDH:FOFFSET:OFFSET? <PORTID>... 104

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:APS <NUMERIC> <NUMERIC> <PORTID>... 105

:SOURCE:DATA:TELECOM:SDH:OVERHEAD:APS? <PORTID> 105

:SOURCE:DATA:TELECOM:SDH:THRUMODE:COVERWRITE <DISCRETE> <PORTID> 106

:SOURCE:DATA:TELECOM:SDH:THRUMODE:COVERWRITE? <PORTID> 106

:SENSE:DATA:TELECOM:SDH:RATE <DISCRETE> <PORTID> 107

:SENSE:DATA:TELECOM:SDH:RATE? <PORTID> 107

:SENSE:DATA:TELECOM:SDH:SCRAMBLING <BOOLEAN> <PORTID> 107

:SENSE:DATA:TELECOM:SDH:SCRAMBLING? <PORTID> 107

:SENSE:DATA:TELECOM:SDH:CHANNELS <STRING> <PORTID> 107

:SENSE:DATA:TELECOM:SDH:CHANNELS? <PORTID> 108

:SENSE:DATA:TELECOM:SDH:CHANNELS:EQUIPPED <LIST> <PORTID> 108

:SENSE:DATA:TELECOM:SDH:CHANNELS:EQUIPPED? <PORTID> 109

:SENSE:DATA:TELECOM:SDH:CHANNELS:UNEQUIPPED <LIST> <PORTID> 109

:SENSE:DATA:TELECOM:SDH:CHANNELS:INVERT <PORTID> 109

:SENSE:DATA:TELECOM:SDH:CHANNELS:ALL <PORTID> 109

:SENSE:DATA:TELECOM:SDH:CHANNELS:AUTO <PORTID> 110

:SENSE:DATA:TELECOM:SDH:CHANNELS:UAUTO <PORTID> 110

:SENSE:DATA:TELECOM:SDH:CHANNELS:AT? <NUMERIC> <PORTID> 110

:SENSE:DATA:TELECOM:SDH:CHANNELS:SPOSITION? <NUMERIC> <PORTID> 110

:SENSE:DATA:TELECOM:SDH:J1:EXPECTED <DISCRETE> <STRING> <CHANNEL> <PORTID> 111

:SENSE:DATA:TELECOM:SDH:J1:EXPECTED? <CHANNEL> <PORTID> 111

:SENSE:DATA:TELECOM:SDH:J1:EXPECTED:LENGTH? <CHANNEL> <PORTID> 111

SENSE:DATA:TELECOM:SDH:J1:EXPECTED:SET:RECEIVED:ALL <PORTHANDLE> 111

SENSE:DATA:TELECOM:SDH:J1:EXPECTED:SET:RECEIVED <PORTHANDLE> <CHANNEL> 111

SENSE:DATA:TELECOM:SDH:J1:EXPECTED:COMPARE:START <PORTHANDLE> 112

SENSE:DATA:TELECOM:SDH:J1:EXPECTED:COMPARE:START? <PORTHANDLE> 112

SENSE:DATA:TELECOM:SDH:J1:EXPECTED:MATCH? <PORTHANDLE> <CHANNEL> 112

SENSE:DATA:TELECOM:SDH:J1:EXPECTED:MATCH:ALL? <PORTHANDLE> 112

:SENSE:DATA:TELECOM:SDH:OCAPTURE:CHANNEL:SELECT <NUMERIC> <PORTID> 112

:SENSE:DATA? <STRING> <PORTID> 114

:SENSE:DATA? <STRING> <PORTID> 115

:SENSE:DATA? <STRING> <PORTID> 115

:SENSE:DATA? <STRING> <PORTID> 115

:SENSE:DATA? <STRING> <PORTID> 116

:SENSE:DATA? <STRING> <PORTID> 116

:SENSE:DATA? <STRING> <PORTID> 117

:FETCH:ARRAY:DATA:TELECOM:SDH:OVERHEAD:DATA? <NUMERIC> <PORTID> 119

:FETCH:SCALAR:DATA:TELECOM:SDH:OVERHEAD:DATA? <NUMERIC> <NUMERIC> <DISCRETE> <PORTID>......119

:FETCH:DATA:TELECOM:SDH:POVERHEAD:MONITOR:CHANNEL <CHANNEL> <PORTID>......120

:FETCH:DATA:TELECOM:SDH:POVERHEAD:MONITOR:CHANNEL? <PORTID> 120

:FETCH:ARRAY:DATA:TELECOM:SDH:POVERHEAD:DATA? <PORTID> 120

:FETCH:SCALAR:DATA:TELECOM:SDH:POVERHEAD:DATA? <DISCRETE> <PORTID>......121

:FETCH:SCALAR:DATA:TELECOM:SDH:OVERHEAD:K1? <PORTID>......121

:FETCH:SCALAR:DATA:TELECOM:SDH:OVERHEAD:K2? <PORTID>......121

:FETCH:STRING:DATA:TELECOM:SDH:J0? <PORTID> 122

:FETCH:STRING:DATA:TELECOM:SDH:J1? <PORTID> 122

:FETCH:STRING:DATA:TELECOM:SDH:J0:LENGTH? <PORTID>......122

:FETCH:STRING:DATA:TELECOM:SDH:J1:LENGTH? <PORTID>......122

:FETCH:ARRAY:DATA:TELECOM:SDH:OCAPTURE? <PORTID>......123

:SOURCE:DATA:TELECOM:SONET:RATE <DISCRETE> <PORTID>......125

:SOURCE:DATA:TELECOM:SONET:RATE? <PORTID> 125

:SOURCE:DATA:TELECOM:SONET:SCRAMBLING <BOOLEAN> <PORTID>......125

:SOURCE:DATA:TELECOM:SONET:SCRAMBLING? <PORTID>......125

:SOURCE:DATA:TELECOM:SONET:PAYLOAD:SCOM <BOOLEAN> <PORTID> 126

:SOURCE:DATA:TELECOM:SONET:PAYLOAD:SCOM ? <PORTID> 126

:SOURCE:DATA:TELECOM:SONET:CHANNELS <STRING> <PORTID> 126

:SOURCE:DATA:TELECOM:SONET:CHANNELS ? <PORTID> 127

:SOURCE:DATA:TELECOM:SONET:CHANNELS:EQUIPPED <STRING> <PORTID> 127

:SOURCE:DATA:TELECOM:SONET:CHANNELS:EQUIPPED ? <PORTID> 127

:SOURCE:DATA:TELECOM:SONET:CHANNELS:UNEQUIPPED <STRING> <PORTID> 128

:SOURCE:DATA:TELECOM:SONET:CHANNELS:INVERT <PORTID> 128

:SOURCE:DATA:TELECOM:SONET:CHANNELS:ALL <PORTID> 128

:SOURCE:DATA:TELECOM:SONET:CHANNELS:AT? <NUMERIC> <PORTID>128

:SOURCE:DATA:TELECOM:SONET:CHANNELS:SPOSITION? <NUMERIC> <PORTID> 129

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:DEFAULT <PORTID> 130

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:DATA <NUMERIC>, <NUMERIC>, <DISCRETE>, <NUMERIC> <PORTID> 130

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:DATA? <NUMERIC>, <NUMERIC> <DISCRETE> <PORTID>.. 131

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:J0:PATTERN <DISCRETE> <PORTID> 131

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:J0:PATTERN? <PORTID> 132

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:J0:PATTERN:B16CRC <STRING> <PORTID> 132

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:J0:PATTERN:B16CRC? <PORTID> 133

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:J0:PATTERN:B64 <STRING> <PORTID> 133

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:J0:PATTERN:B64? <PORTID> 133

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:DEFAULT 133

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:DATA <DISCRETE>,<NUMERIC> <CHANNEL> <PORTID> 133

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:DATA? <DISCRETE> <CHANNEL> <PORTID> 134

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:J1:PATTERN <DISCRETE> <CHANNEL> <PORTID> 134

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:J1:PATTERN? <CHANNEL> <PORTID> 135

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:J1:PATTERN:B16CRC <STRING> <CHANNEL> <PORTID> 135

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:J1:PATTERN:B16CRC? <CHANNEL> <PORTID> 136

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:J1:PATTERN:B64 <STRING> <CHANNEL> <PORTID> 136

:SOURCE:DATA:TELECOM:SONET:POVERHEAD:J1:PATTERN:B64? <CHANNEL> <PORTID> 136

:SOURCE:DATA:TELECOM:SONET:ERROR:CHANNELS:SELECTED <STRING> <PORTID> 137

:SOURCE:DATA:TELECOM:SONET:ERROR:CHANNELS:SELECTED? <PORTID> 137

:SOURCE:DATA:TELECOM:SONET:ERROR:CHANNELS:UNSELECTED <STRING> <PORTID> 137

:SOURCE:DATA:TELECOM:SONET:ERROR:CHANNELS:UNSELECTED? <PORTID>......138

:SOURCE:DATA:TELECOM:SONET:ERROR:CHANNELS:INVERT <PORTID>. 138

:SOURCE:DATA:TELECOM:SONET:ERROR:CHANNELS:ALL <PORTID>........138

:SOURCE:DATA:TELECOM:SONET:ERROR:GROUP <DISCRETE> <PORTID> 138

:SOURCE:DATA:TELECOM:SONET:ERROR:GROUP? <PORTID>......................138

:SOURCE:DATA:TELECOM:SONET:ERROR:SECTION <DISCRETE> <PORTID> 139

:SOURCE:DATA:TELECOM:SONET:ERROR:SECTION? <PORTID> 139

:SOURCE:DATA:TELECOM:SONET:ERROR:SECTION:REIL <DISCRETE> <PORTID>......139

:SOURCE:DATA:TELECOM:SONET:ERROR:SECTION:REIL? <PORTID>..........139

:SOURCE:DATA:TELECOM:SONET:ERROR:PATH <DISCRETE> <PORTID> .. 139

:SOURCE:DATA:TELECOM:SONET:ERROR:PATH? <PORTID>..........................140

:SOURCE:DATA:TELECOM:SONET:ERROR:PAYLOAD <DISCRETE> 140

:SOURCE:DATA:TELECOM:SONET:ERROR:PAYLOAD?......................................140

:SOURCE:DATA:TELECOM:ERROR:SINGLE <PORTID>......................................140

:SOURCE:DATA:TELECOM:SONET:ERROR:RATE:USER <DISCRETE> <NUMERIC> <DISCRETE> <PORTID>......141

:SOURCE:DATA:TELECOM:SONET:ERROR:RATE:USER? <DISCRETE> <PORTID>......142

:SOURCE:DATA:TELECOM:SONET:ERROR:RATE:USER:ACTION <BOOLEAN> <PORTID>......142

:SOURCE:DATA:TELECOM:SONET:ERROR:RATE:USER:ACTION:ALL <BOOLEAN> 142

:SOURCE:DATA:TELECOM:SONET:ALARM:GROUP <DISCRETE> <PORTID>143

:SOURCE:DATA:TELECOM:SONET:ALARM:GROUP? <PORTID>.......... 143

:SOURCE:DATA:TELECOM:SONET:ALARM:SECTION <DISCRETE> <PORTID> 143

:SOURCE:DATA:TELECOM:SONET:ALARM:SECTION? <PORTID> 144

:SOURCE:DATA:TELECOM:SONET:ALARM:SEF <PORTID> 144

:SOURCE:DATA:TELECOM:SONET:ALARM:PATH <DISCRETE> <PORTID> .. 144

:SOURCE:DATA:TELECOM:SONET:ALARM:PATH? <PORTID>.......... 144

:SOURCE:DATA:TELECOM:SONET:ALARM:PATH:ERDIP:MODE <BOOLEAN> <PORTID> 144

:SOURCE:DATA:TELECOM:SONET:ALARM:PATH:ERDIP:MODE? <PORTID>145

:SOURCE:DATA:TELECOM:SONET:ALARM:PATH:VALUE <DISCRETE> <NUMERIC> <PORTID> 145

:SOURCE:DATA:TELECOM:SONET:ALARM:PATH:VALUE? <DISCRETE> <PORTID> 145

:SOURCE:DATA:TELECOM:SONET:ALARM:PATH:VALUE:DEFAULT <DISCRETE> <PORTID>.......... 145

:SOURCE:DATA:TELECOM:SONET:POINTER:VALUE <NUMERIC> <BOOLEAN> <PORTID>.......... 146

:SOURCE:DATA:TELECOM:SONET:POINTER:VALUE? <PORTID> 146

:SOURCE:DATA:TELECOM:SONET:POINTER:VALUE:INCREMENT <PORTID> 146

:SOURCE:DATA:TELECOM:SONET:POINTER:VALUE:DECREMENT <PORTID> 146

:SOURCE:CLOCK:SONET:FOFFSET:OFFSET <NUMERIC><PORTID>............... 147

:SOURCE:CLOCK:SONET:FOFFSET:OFFSET? <PORTID>.................................... 147

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:APS <NUMERIC> <NUMERIC> <PORTID>... 148

:SOURCE:DATA:TELECOM:SONET:OVERHEAD:APS? <PORTID> 148

:SOURCE:DATA:TELECOM:SONET:THRUMODE:COVERWRITE <BOOLEAN> <PORTID>... 149

:SOURCE:DATA:TELECOM:SONET:THRUMODE:COVERWRITE? <PORTID>. 149

:SENSE:DATA:TELECOM:SONET:RATE <DISCRETE> <PORTID>..................... 150

:SENSE:DATA:TELECOM:SONET:RATE? <PORTID>... 150

:SENSE:DATA:TELECOM:SONET:SCRAMBLING <BOOLEAN> <PORTID> 150

:SENSE:DATA:TELECOM:SONET:SCRAMBLING? <PORTID>............................ 150

:SENSE:DATA:TELECOM:SONET:PAYLOAD:SCOM <BOOLEAN> <PORTID> 150

:SENSE:DATA:TELECOM:SONET:PAYLOAD:SCOM ? <PORTID>...................... 151

:SENSE:DATA:TELECOM:SONET:CHANNELS <STRING> <PORTID>............... 151

:SENSE:DATA:TELECOM:SONET:CHANNELS ? <PORTID>................................ 152

:SENSE:DATA:TELECOM:SONET:CHANNELS:EQUIPPED <STRING> <PORTID> 152

:SENSE:DATA:TELECOM:SONET:CHANNELS:EQUIPPED ? <PORTID>............ 152

:SENSE:DATA:TELECOM:SONET:CHANNELS:UNEQUIPPED <STRING> <PORTID>... 153

:SENSE:DATA:TELECOM:SONET:CHANNELS:INVERT <PORTID>.................. 153

:SENSE:DATA:TELECOM:SONET:CHANNELS:ALL <PORTID>.......................... 153

:SENSE:DATA:TELECOM:SONET:CHANNELS:AUTO <PORTID>...................... 153

:SENSE:DATA:TELECOM:SONET:CHANNELS:UAUTO <PORTID>.................... 153

:SENSE:DATA:TELECOM:SONET:CHANNELS:AT? <NUMERIC> <PORTID>... 153

:SENSE:DATA:TELECOM:SONET:CHANNELS:SPOSITION? <NUMERIC> <PORTID>.. 154

:SENSE:DATA:TELECOM:SONET:J1:EXPECTED <DISCRETE> <STRING> <CHANNEL> <PORTID>... 154

:SENSE:DATA:TELECOM:SONET:J1:EXPECTED? <CHANNEL> <PORTID>.... 154

:SENSE:DATA:TELECOM:SONET:J1:EXPECTED:LENGTH? <CHANNEL> <PORTID>.. 155

SENSE:DATA:TELECOM:SONET:J1:EXPECTED:SET:RECEIVED:ALL <PORTHANDLE>... 155

SENSE:DATA:TELECOM:SONET:J1:EXPECTED:SET:RECEIVED <PORTHANDLE> <CHANNEL> 155

SENSE:DATA:TELECOM:SONET:J1:EXPECTED:COMPARE:START <PORTHANDLE>... 155

SENSE:DATA:TELECOM:SONET:J1:EXPECTED:COMPARE:START? <PORTHANDLE>... 155

SENSE:DATA:TELECOM:SONET:J1:EXPECTED:MATCH? <PORTHANDLE> <CHANNEL> 155

SENSE:DATA:TELECOM:SONET:J1:EXPECTED:MATCH:ALL? <PORTHANDLE>... 156

:SENSE:DATA:TELECOM:SONET:OCAPTURE:STS1 <NUMERIC> <PORTID> . 156

:SENSE:DATA:TELECOM:SONET:OCAPTURE:STS1? <PORTID> 156

:SENSE:DATA:TELECOM:SONET:OCAPTURE:STS3:SELECT <NUMERIC> <PORTID> 156

:SENSE:DATA? <STRING> <PORTID> 158

:SENSE:DATA? <STRING> <PORTID> 159

:SENSE:DATA? <STRING> <PORTID> 159

:SENSE:DATA? <STRING> <PORTID> 159

:SENSE:DATA? <STRING> <PORTID> 160

:SENSE:DATA? <STRING> <PORTID> 160

:SENSE:DATA? <STRING> <PORTID> 161

:FETCH:ARRAY:DATA:TELECOM:SONET:OVERHEAD:DATA? <NUMERIC> <PORTID> 162

:FETCH:SCALAR:DATA:TELECOM:SONET:OVERHEAD:DATA? <NUMERIC> <NUMERIC> <DISCRETE> <PORTID> 162

:FETCH:DATA:TELECOM:SONET:POVERHEAD:MONITOR:CHANNEL <CHANNEL> <PORTID> 163

:FETCH:DATA:TELECOM:SONET:POVERHEAD:MONITOR:CHANNEL? <PORTID> 163

:FETCH:ARRAY:DATA:TELECOM:SONET:POVERHEAD:DATA? <PORTID> ..163

:FETCH:SCALAR:DATA:TELECOM:SONET:POVERHEAD:DATA? <DISCRETE> <PORTID> 163

:FETCH:ARRAY:DATA:TELECOM:SONET:OCAPTURE? <PORTID>..................164

:FETCH:SCALAR:DATA:TELECOM:SONET:OVERHEAD:K1? <PORTID> 164

:FETCH:SCALAR:DATA:TELECOM:SONET:OVERHEAD:K2? <PORTID> 164

:FETCH:STRING:DATA:TELECOM:SONET:J0? <PORTID>....................................165

:FETCH:STRING:DATA:TELECOM:SONET:J1? <PORTID>....................................165

:FETCH:STRING:DATA:TELECOM:SONET:J0:LENGTH? <PORTID>..................165

:FETCH:STRING:DATA:TELECOM:SONET:J1:LENGTH? <PORTID>..................165

:STATUS:<STATUS REGISTER>:ENABLE <NUMERIC>.......................................166

:STATUS:CHISTORY <PORTID>...167

STATUS:QUESTIONABLE:CONDITION?...174

STATUS:QUESTIONABLE:ENABLE <NUMERIC>...174

STATUS:QUESTIONABLE:ENABLE?...174

STATUS:QUESTIONABLE:EVENT?..174

STATUS:QUESTIONABLE:NTRANSITION <NUMERIC>......................................174

STATUS:QUESTIONABLE:NTRANSITION?...174

STATUS:QUESTIONABLE:PTRANSITION <NUMERIC>.......................................174

STATUS:QUESTIONABLE:PTRANSITION?..174

STATUS:OPERATIONAL:CONDITION?...176

STATUS:OPERATIONAL:ENABLE <NUMERIC> 176

STATUS:OPERATIONAL:ENABLE? 176

STATUS:OPERATIONAL:EVENT?...176

STATUS:OPERATIONAL:NTRANSITION <NUMERIC> 176

STATUS:OPERATIONAL:NTRANSITION? 176

STATUS:OPERATIONAL:PTRANSITION <NUMERIC>...176

STATUS:OPERATIONAL:PTRANSITION? INSTRUMENT STATUS REGISTER 176

INSTRUMENT STATUS REGISTER 177

STATUS:INSTRUMENT:CONDITION? 178

STATUS:INSTRUMENT:ENABLE <NUMERIC> 178

STATUS:INSTRUMENT:ENABLE? 178

STATUS:INSTRUMENT:EVENT? 178

STATUS:INSTRUMENT:NTRANSITION <NUMERIC> 178

STATUS:INSTRUMENT:NTRANSITION? 178

STATUS:INSTRUMENT:PTRANSITION <NUMERIC> 178

STATUS:INSTRUMENT:PTRANSITION? DATA STATUS REGISTER 178

DATA STATUS REGISTER 179

STATUS:DATA:CONDITION? 180

STATUS:DATA:ENABLE <NUMERIC> 180

STATUS:DATA:ENABLE? 180

STATUS:DATA:EVENT? 180

STATUS:DATA:NTRANSITION <NUMERIC> 180

STATUS:DATA:NTRANSITION? 180

STATUS:DATA:PTRANSITION <NUMERIC> 180

STATUS:DATA:PTRANSITION? SONET | SDH PATH STATUS REGISTER 180

SONET | SDH PATH STATUS REGISTER..181

:STATUS:SONET:PATH:CONDITION? <<CHANNEL>>, <PORTID>....................182

:STATUS:SONET:PATH:ENABLE <NUMERIC>, <<CHANNEL>>, <PORTID> ...182

:STATUS:SONET:PATH:ENABLE? <CHANNEL>, <PORTID>...............................182

:STATUS:SONET:PATH:EVENT? <CHANNEL>, <PORTID> 182

:STATUS:SONET:PATH:NTRANSITION <NUMERIC>, <CHANNEL>, <PORTID> 182

:STATUS:SONET:PATH:NTRANSITION? <CHANNEL>, <PORTID>....................182

:STATUS:SONET:PATH:PTRANSITION <NUMERIC>, <CHANNEL>, <PORTID> 182

:STATUS:SONET:PATH:PTRANSITION? <CHANNEL>, <PORTID> 182

:STATUS:SONET:PATH:HISTORY? <CHANNEL>, <PORTID>SONET | SDH:PATH:SUMMARY STATUS REGISTER 182

SONET | SDH:PATH:SUMMARY STATUS REGISTER.........................183

:STATUS:SONET:PATH:SUMMARY:CONDITION? <PORTID>............................184

:STATUS:SONET:PATH:SUMMARY:ENABLE <NUMERIC>, <PORTID>........184

:STATUS:SONET:PATH:SUMMARY:ENABLE? <PORTID>..............................184

:STATUS:SONET:PATH:SUMMARY:EVENT? <PORTID> 184

:STATUS:SONET:PATH:SUMMARY:NTRANSITION <NUMERIC>, <PORTID> 184

:STATUS:SONET:PATH:SUMMARY:NTRANSITION? <PORTID> 184

:STATUS:SONET:PATH:SUMMARY:PTRANSITION <NUMERIC>, <PORTID> 184

:STATUS:SONET:PATH:SUMMARY:PTRANSITION? <PORTID>........................184

:STATUS:SONET:PATH:SUMMARY:HISTORY?   <PORTID> 184

:STATUS:SONET:PORT:CONDITION? <PORTID> 187

:STATUS:SONET:PORT:ENABLE <NUMERIC>, <PORTID> 187

:STATUS:SONET:PORT:ENABLE? <PORTID> 187

:STATUS:SONET:PORT:EVENT? <PORTID> 187

:STATUS:SONET:PORT:NTRANSITION <NUMERIC>, <PORTID> 187

:STATUS:SONET:PORT:NTRANSITION? <PORTID> 187

:STATUS:SONET:PORT:PTRANSITION <NUMERIC>, <PORTID> 187

:STATUS:SONET:PORT:PTRANSITION? <PORTID> 187

:STATUS:SONET:PORT:HISTORY? <PORTID> 187

:STATUS:SONET:PORT:SUMMARY:CONDITION? 189

:STATUS:SONET:PORT:SUMMARY:ENABLE <NUMERIC> 189

:STATUS:SONET:PORT:SUMMARY:ENABLE? 189

:STATUS:SONET:PORT:SUMMARY:EVENT? 189

:STATUS:SONET:PORT:SUMMARY:NTRANSITION <NUMERIC>, ... 189

:STATUS:SONET:PORT:SUMMARY:NTRANSITION? 189

:STATUS:SONET:PORT:SUMMARY:PTRANSITION <NUMERIC>, ... 189

:STATUS:SONET:PORT:SUMMARY:PTRANSITION? 189

:STATUS:SONET:PORT:SUMMARY:HISTORY? 189

Printed in U.K. 07/03
J7241-90002

Agilent Technologies