Programmer Manual

Tektronix

CTS 710
SONET Test Set

070-8924-02

This document applies to firmware version 1.30
and above.

Tektronix, Inc., P.O. Box **500 Beaverton, OR. 97077**

WARRANTY

Tektronix warrants that this product will be free from defects in materials and workmanship for a period of three (3) years from the date of shipment. If any such product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; or c) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THIS PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESSED OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

Getting Started

Syntax and Commands

Status and Events

Examples

Appendices

Glossary and Index

List of Figures

List of Tables

General Safety Summary

Review the following safety precautions to avoid injury and prevent damage to this product or any products connected to it.

Only qualified personnel should perform service procedures.

While using this product, you may need to access other parts of the system. Read the *General Safety Summary* in other system manuals for warnings and cautions related to operating the system.

Injury Precautions

Use Proper Power Cord

To avoid fire hazard, use only the power cord specified for this product.

Avoid Electric Overload

To avoid electric shock or fire hazard, do not apply a voltage to a terminal that is outside the range specified for that terminal.

Ground the Product

This product is grounded through the grounding conductor of the power cord. To avoid electric shock, the grounding conductor must be connected to earth ground. Before making connections to the input or output terminals of the product, ensure that the product is properly grounded.

Do Not Operate Without Covers

To avoid electric shock or fire hazard, do not operate this product with covers or panels removed.

Use Proper Fuse

To avoid fire hazard, use only the fuse type and rating specified for this product.

Do Not Operate in Wet/Damp Conditions

To avoid electric shock, do not operate this product in wet or damp conditions.

Do Not Operate in Explosive Atmosphere

To avoid injury or fire hazard, do not operate this product in an explosive atmosphere.

Wear Eye Protection

To avoid eye injury, wear eye protection if there is a possibility of exposure to high-intensity rays.

Product Damage Precautions

Use Proper Power Source

Do not operate this product from a power source that applies more than the voltage specified.

Provide Proper Ventilation

To prevent product overheating, provide proper ventilation.

Do Not Operate With Suspected Failures

If you suspect there is damage to this product, have it inspected by qualified service personnel.

Safety Terms and Symbols

Terms in This Manual

These terms may appear in this manual:

WARNING. *Warning statements identify conditions or practices that could result in injury or loss of life.*

CAUTION. *Caution statements identify conditions or practices that could result in damage to this product or other property.*

Terms on the Product

These terms may appear on the product:

DANGER indicates an injury hazard immediately accessible as you read the marking.

WARNING indicates an injury hazard not immediately accessible as you read the marking.

CAUTION indicates a hazard to property including the product.

Symbols on the Product

The following symbols may appear on the product:

DANGER
High Voltage

Protective Ground
(Earth) Terminal

ATTENTION
Refer to
Manual

Double
Insulated

Certifications and Compliances

CSA Certified Power Cords

CSA Certification includes the products and power cords appropriate for use in the North America power network. All other power cords supplied are approved for the country of use.

Preface

The CTS 710 SONET Test Set can be controlled remotely through the use of SCPI-derived commands (Standard Commands for Programmable Instruments) and IEEE 488.2 Common Commands. This manual describes how to use these commands to access information generated by or stored in the instrument.

About This Manual

The following table shows you where to find information on the more common subjects related to programming your CTS 710.

Where To Find Information in This Manual

If you want to know:	Look in this part of the manual:
How to set up the instrument for remote communication	*Setting Up for Remote Communication* starting on page 1–1
How the programming model for this instrument is structured	*Programming Model* starting on page 1–7
How to perform simple tasks such as generating a normal or modified signal	*Examples of Command Usage* starting on page 1–13
How the command language syntax is structured	*Syntax* starting on page 2–1
What the functional command groups are	*Functional Command Groups* starting on page 2–7
What the commands and queries are	*Transmit Commands* section on page 2–11 through the *Common Commands* section
What the error and event messages are	*Status and Events* section contains the primary error and event messages starting on page 3–8 The commands and queries in *Syntax and Commands* starting on page 2–11 list the primary and secondary error and event messages
How to structure a program containing commands and queries	*Examples* starting on page 4–1
What the default values for the instrument are	*Appendix D*

Conventions

The procedures in the *Setting Up for Remove Communication* section that require the use of the front panel are presented in table format. Perform the procedures by reading from left to right in the table (see example below). The word *none* in a cell indicates that no action is required.

Press Menu Button	Select Menu Page	Highlight Parameter	Select Choice
Begin here with Step 1	Step 2	Step 3	Step 4
		Step 5	Step 6
		Step 7	Step 8, CTS 710 instruction is complete

Some procedures require several iterations of highlighting parameters and selecting choices. Some procedures may require more than one menu button or menu page selection as well.

Related Manuals

The following documents are also available for the CTS 710 SONET Test Set.

- The *CTS 710 User Manual* (Tektronix part number 070-8852-XX) is the primary source of information about how the CTS 710 functions.
- The *CTS 710 SONET Test Set & CTS 750 SDH Test Set Reference* (Tektronix part number 070-8854-XX) provides a quick overview of the instrument programming commands.
- The *CTS 710 SONET Test Set Reference* (Tektronix part number 070-9336-XX) provides a quick overview of the menu structure and alarms.
- The *CTS 710 SONET Test Set & CTS 750 SDH Test Set Service Manual* (Tektronix part number 070-8853-XX) provides information on maintaining and servicing your instrument to the module level.

Setting Up for Remote Communication

With a computer (controller), you can control the CTS 710 over GPIB or RS-232. This section shows you how to do the following:

- Connect the instrument to the computer
- Set the communication parameters
- Test the communication

NOTE. *In order for data communication to take place, the computer should have the proper communication hardware (either GPIB or RS-232) and software drivers already installed. Consult the manufacturer's manuals for detailed information.*

GPIB Connection

To connect the CTS 710 to a computer via GPIB, perform the following procedure:

1. Locate the GPIB port on the rear-panel (see Figure 1–1).
2. Connect one end of the GPIB cable to the CTS 710 and the other end to the computer.
3. Connect both the CTS 710 and the computer to the line voltage.
4. Turn on the instrument.
5. Turn on the computer.

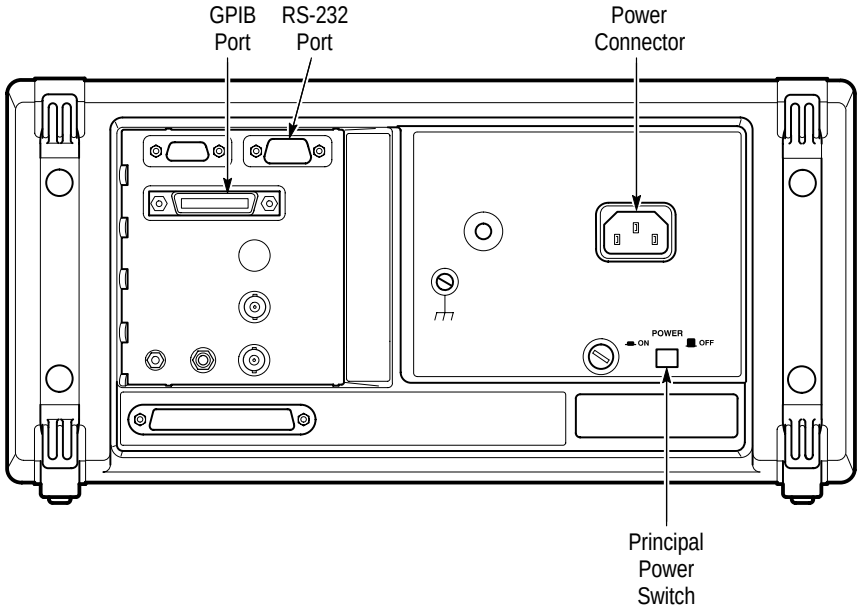


Figure 1–1: Rear-Panel Communication Ports

GPIB Parameter Setup

The only parameter you may need to set is the GPIB address; the default address set at the factory is 4. To change the GPIB address to other values, perform the steps indicated in Table 1–1.

Table 1–1: Setting GPIB Address

Press Menu Button	Select Menu Page	Highlight Parameter	Select Choice
UTILITY	REMOTE CONTROL	GPIB Primary Address	Inc, Dec, or Offline

Figure 1–2 is a display of the UTILITY Menu showing the GPIB Address.

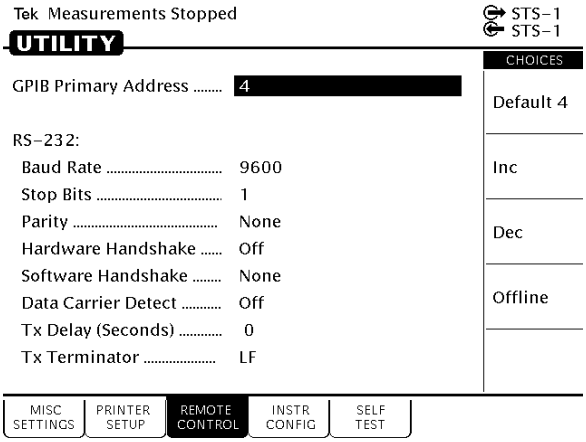


Figure 1–2: GPIB Address in the UTILITY Menu

GPIB Connection Test

To test whether the GPIB connection is working, send a GPIB query from the computer. The *Examples* section shows how to send the *IDN? query command to check the identity of the instrument. The first two terms of the response should be “TEKTRONIX” and “CTS 710”, respectively.

RS-232 Connection

The RS-232 port of the CTS 710 is a DB-9 male connector. Table 1–2 lists the pin assignments of the connector.

Table 1–2: RS-232 Pin Assignments

Pin Number	Description
1	Data Carrier Detect
2	Receive Data
3	Transmit Data
4	Data Terminal Ready
5	Signal Ground
6	Data Set Ready
7	Request to Send

Table 1–2: RS-232 Pin Assignments (Cont.)

Pin Number	Description
8	Clear to Send
9	Ring Indicator

The instrument is configured as a DTE (Data Terminal Equipment).

To connect the instrument to a computer via RS-232:

1. Locate the RS-232 port on the rear-panel (see Figure 1–1).
2. Connect one end of the RS-232 cable to the instrument and the other end to the computer.
3. Connect both the CTS 710 and the computer to the power system.
4. Turn on the instrument.
5. Turn on the computer.

***NOTE**. If you use the RS-232 port for the computer, you cannot use it for the printer.*

RS-232 Parameter Setup

Table 1–3 shows the default RS-232 parameter values.

Table 1–3: Default RS-232 Parameter Values

Parameter	Default Value
Baud Rate	9600
Stop Bits	1
Parity	None
Hardware Handshake	Off
Software Handshake	None
Data Carrier Detect	Off
Tx Delay (seconds)	0
Tx Terminator	LF

To change the RS-232 parameters to other values, perform the steps indicated in Table 1–4.

Table 1–4: Setting RS-232 Parameters

Press Menu Button	Select Menu Page	Highlight Parameter	Select Choice
UTILITY	REMOTE CONTROL	Baud Rate	1200, 2400, 4800, or 9600
		Stop Bits	1 or 2
		Parity	None, Odd, or Even
		Hardware Handshake	Off or On
		Software Handshake	None or Xon/Xoff
		Tx Delay (seconds)	0, 1, 5, Inc, or Dec
		Tx Terminator	LF, CR, CR/LF, or LF/CR

RS-232 Connection Test

To verify that the RS-232 connection works, launch a terminal emulation program on the computer. Configure the terminal settings to be the same values as those you have selected on the CTS 710. Turn on the instrument, and a one-line identification message should appear on the computer screen.

If the message does not appear or is unreadable, it is an indication that connection is not set up properly. Table 1–5 shows some common problems and their possible solutions.

Table 1–5: RS-232 Connection Problems

Symptom	Probable Cause	Solution
No communication	Defective cable	Replace the cable
Garbled characters	Incompatible baud setting	Set the RS-232 parameters on computer to be the same as those on the instrument
Missing characters	Incorrect flow control	Use a null modem or set the software handshaking to Xon/Xoff

Programming Model

This section explains the two functional blocks of the CTS 710 SONET Test Set, the subsystems of each functional block, and the two modes in which the instrument operates.

Functional Blocks

The CTS 710 is made up of two independent functional blocks: a Transmitter and a Receiver. The Transmitter generates a SONET signal of known characteristics for testing. The Receiver accepts a telecommunications signal, breaks it apart to see what has gone wrong or been modified, and displays measurements for that signal.

Transmitter

The Transmitter consists of two major subsystems: OUTPUT1 and SOURCE. The OUTPUT1 subsystem commands determine *how* the signal is transmitted and set characteristics such as the signal type and transmission rate. The SOURCE subsystem commands determine *what* signal is transmitted. Use the SOURCE subsystem commands to set errors, alarms, failures, pointers, overhead, and the payload. These two subsystems correspond to the functions in the **TRANSMIT** menu of your instrument.

Receiver

The Receiver is made up of two major subsystems: INPUT1 and SENSE. The INPUT1 subsystem commands determine *how* a signal is received and set characteristics such as the signal type and rate. The SENSE subsystem commands determine *what* signal is received. Use the SENSE subsystem commands to set up tests, capture overhead, and access measurements. These two subsystems correspond to the functions in the **RECEIVE** and **RESULTS** menus of your instrument.

Modes of Operation

The CTS 710 operates in normal or through mode. Figure 1–3 illustrates how the modes of operation work within the two functional blocks.

Normal Mode

Use normal mode to generate a signal of known characteristics and to measure a received signal. You can generate a normal or modified signal in this mode. The SOURce:DATA:TELecom:SOURce OUTPUT1 command instructs the CTS 710 to operate in normal mode.

Through Mode

Use through mode to transmit a received signal without modifying it. The rate and content of the transmitted signal are matched to that of the received signal. The SOURce:DATA:TELecom:SOURce INPUT1 command instructs the CTS 710 to operate in through mode; the instrument continues to measure the received signal.

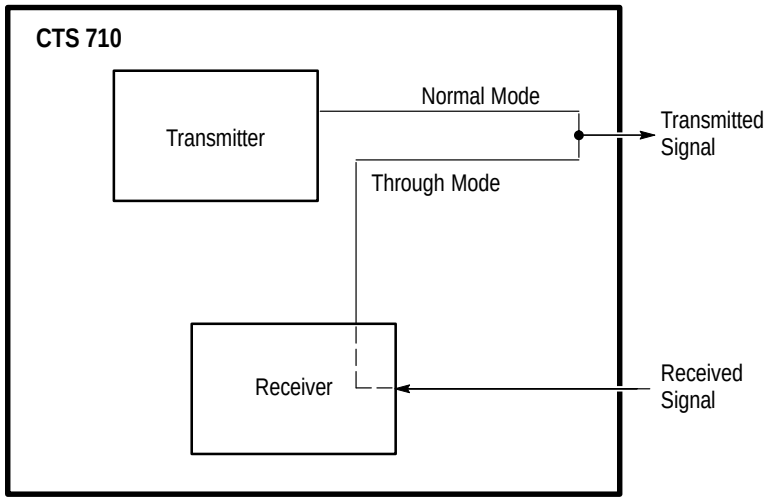


Figure 1–3: Modes of Operation

Information Storage in the Instrument

Instrument setups, pass/fail tests, and test results are stored in several types of internal buffers and on floppy disk. The number and type of buffers available depend on the type of information stored. The following sections detail the storage locations available for setups, tests, and results.

Storage of Instrument Setups

Instrument setups contain all information necessary to set the instrument operation. These setups are stored in internal buffers and on floppy disk as shown in Figure 1–4. The lines and arrows in the illustration show the direction that information is moved. The commands you use to move the setups from one location to another are shown near the lines.

The current buffer (buffer number 0) contains the current instrument setups. Buffers 1 through 5 contain additional instrument setups. The floppy disk can contain any instrument setups.

You can move information between the current buffer and buffers 1 through 5 by using the *SAV and *RCL commands. Specify the buffer number with these commands. For example, use the *SAV 3 command to save the current buffer

information in buffer 3 and the *RCL 3 command to move information from buffer 3 to the current buffer.

You can also move instrument setups between the floppy disk and the buffers. Use the MMEMory:STORe:SETTings command to move information from the buffers to the floppy disk. Specify the file name and buffer number with this command (the current buffer is buffer number 0). For example, use the MMEMory:STORe:SETTings “SET001”,2 command to store the information in buffer number 2 to a file named SET001. Use the MMEMory:STORe:SETTings “SET001”,0 command to store information in the current buffer to a disk file.

Use the MMEMory:LOAD:SETTings command to move information from the disk to the buffers. For example, use the MMEMory:LOAD:SETTings “SET001”,4 command to move information from a file named SET001 on disk to buffer 4. Use the MMEMory:LOAD:SETTings “SET001”,0 command to move information from a disk file to the current buffer.

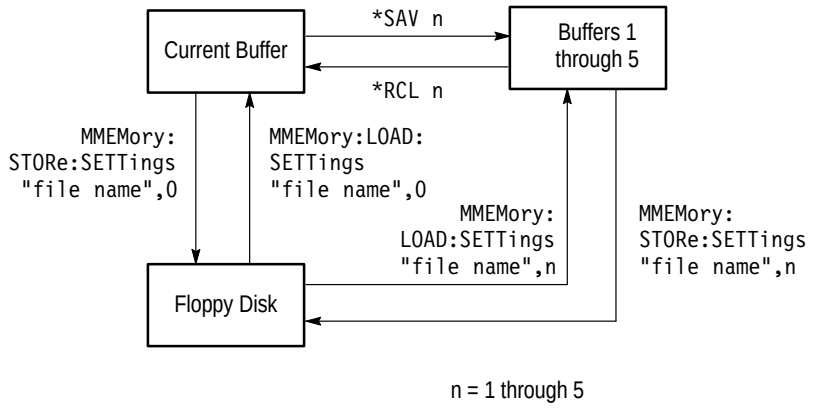


Figure 1–4: How Instrument Setups Are Stored

Storage of Pass/Fail Tests

Pass/fail tests allow you to apply predefined criteria to test results and determine if the tests passed or failed. The pass/fail tests are stored in a current buffer and on floppy disk as shown in Figure 1–5. The lines and arrows in the illustration show the direction that information is moved. The commands you use to move the pass/fail tests from one location to another are shown near the lines. The current buffer contains the current pass/fail test. The floppy disk can contain any pass/fail tests.

Use the MMEMory:STORe:TEST command to move information from the current buffer to the disk. Specify the file name with this command. For example, the MMEMory:STORe:TEST “TEST001” command will store the current pass/fail test in a file named TEST001 on the disk.

Use the MMEMory:LOAD:TEST command to move information from the disk to the current buffer. Specify the file name with this command. For example, the

MMEMory:LOAD:TEST “TEST002” will copy the pass/fail test in a file named TEST002 on disk to the current buffer.

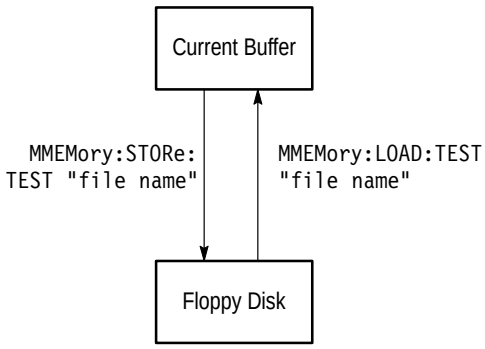


Figure 1–5: How Pass/Fail Tests Are Stored

Storage of Test Results

Test results contain measurement obtained from the Receiver and are initiated by the SENSe:DATA:TELecom:TEST:STARt, SENSe:DATA:TELecom:TEST:STOP, and SENSe:DATA:TELecom:TEST:DURation commands. Test results are stored in a different way than the instrument setups and pass/fail tests. The test results are stored in a current and previous buffer, on floppy disk, and in two additional buffers: buffer number –1 and the view buffer (see Figure 1–6). Buffer number –1 contains only test results that have been copied from disk. The view buffer is a view-only buffer that contains test results from any buffer or from disk. Note that information can be copied only to the view buffer and not from it. The lines in the illustration show the direction that information is moved. The commands you use to move the test results from one location to another are shown near these lines.

The view buffer and disk can contain current or previous test results. Buffer 1 contains current test results, buffer 2 contains previous test results, and buffer number –1 contains test results from disk.

You can move information to the view buffer in three ways. The MMEMory:LOAD:RESUlts moves test results from the disk to this buffer (test results are automatically moved to buffer number –1 when you send this command). The SENSe:DATA:TELecom:MEASure:BUFFer 2 command moves test results from buffer 2 (previous test results) to the view buffer. And the SENSe:DATA:TELecom:MEASure:BUFFer 1 command moves test results from buffer 1 (current test results) to the view buffer. When the SENSe:DATA:TELecom:TEST:STARt command is sent, the test results in buffer 1 are automatically moved to the view buffer.

You can move information from buffers 1 and 2 to disk by using the MMEMory:STORe:RESUlts command. Specify the file name and buffer number 1 or 2 with this command. For example, the MMEMory:STORe:RESUlts “RES001”, 2

command stores the test results from buffer 2 to a file named RES001 on the disk.

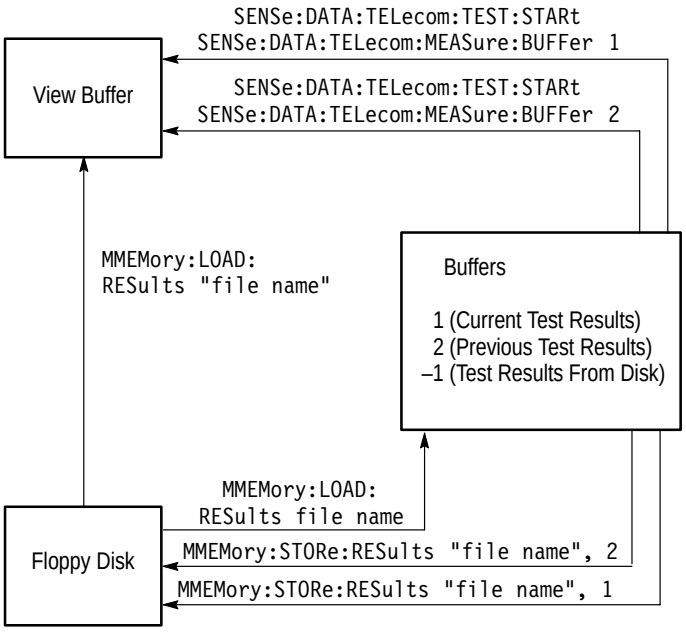


Figure 1–6: How Test Results Are Stored

Examples of Command Usage

This section shows you how to use the commands and queries together to do such things as setting up normal or modified signals, generating errors, and accessing measurements. Use this section to learn about the command language before you start writing programs. Be sure to refer to the *Syntax and Commands* section for more details on how to construct commands and queries correctly.

NOTE. *The commands and queries in this section are shown as a combination of uppercase and lowercase letters. The uppercase letters signify the accepted abbreviation of the command or query.*

Generating Signals

This section shows you how to generate normal and modified signals.

Generating Signals in Normal Mode

The following example shows you how to set up a normal electrical signal at an STS-1 rate, using channel one, and which contains a PRBS 2E23–1 test pattern:

1. Reset the CTS 710 to a known state by sending the *RST command. *Appendix D* lists the default parameter values.
2. Set up the signal physical characteristics by sending the following commands:
   OUTPUT1:TELecom:RATE STS1
   OUTPUT1:TELecom:TYPE ELECtrical
   OUTPUT1:TELecom:LEVel XCON
3. Set the instrument to normal mode by sending the SOURce:DATA:TELecom:SOURce OUTPUT1 command.
4. Set up to test on channel one by sending the SOURce:DATA:TELecom:CHANnel 1 command.
5. Select mapping by sending the SOURce:DATA:TELecom:PAYLoad:MAPPing EQUIpped command; this command sets the C2 byte to 01.
6. Select a payload test pattern of PRBS 2E23–1 by sending the SOURce:DATA:TELecom:PAYLoad:PATTern PRBS23 command.

At this point the instrument is generating a normal signal. You can now modify the signal or insert errors or alarms; the *Generating Modified Signals* section shows you how.

Generating Signals in Through Mode

When the instrument is operating in through mode, the received signal is transmitted without modification.

Set the instrument to through mode by sending the SOURce:DATA:TELecom: SOURce INPUT1 command.

Generating Modified Signals

This section shows you a few of the many ways to create modified signals.

Changing the Overhead. Use the following methods to insert information into the overhead to create a modified signal:

- Insert overhead data into the Section DCC (bytes D1 and D2) from an external protocol analyzer by sending the SOURce:DATA:TELecom: OVERhead:INSert SDCC command. You can also insert data into the line DCC (bytes D4 through D12) or the F1 byte by using LDCC or F1 as the parameter value.
- Change the A1 byte on channel one by sending the SOURce:DATA: TELecom:OVERhead:DATA 1,A1,0,246 command. This command sets the A1 byte to a value of 246 (binary 11110110). The table included with the command description on page 2–40 lists the bytes available for selection.
- You can change the APS bytes in two ways:
  - Send the SOURce:DATA:TELecom:OVERhead:DATA 1,K1,0,#H0 command to set the K1 byte to zero. Use this command in the same way to set the K2 byte.
  - Send the SOURce:DATA:TELecom:OVERhead:APS command to set the K1 and K2 bytes together. For example, sending the SOURce:DATA: TELecom:OVERhead:APS #HFFFF command sets both K1 and K2 to decimal 255 (all ones in binary).

Changing Pointers. Use one of the following methods to adjust pointers.

To create a manual pointer adjustment, follow these steps:

1. Send the SOURce:DATA:TELecom:POINter:MODE MANual command.
2. Send the SOURce:DATA:TELecom:POINter:VALue 590 command to create a pointer with a value 590. (If you use a value greater than 782, an illegal pointer is created.)
3. To generate a new data flag every time a pointer changes, send the SOURce:DATA:TELecom:POINter:NDFLag ON command.

To create pointer adjustments that alternately increment and decrement, follow these steps:

1. Send the SOURce:DATA:TELecom:POINter:MODE SINGle command.
2. Send the SOURce:DATA:TELecom:POINter:ACTion command to initiate the pointer adjustment.

To create a burst of pointer adjustments, follow these steps:

1. Send the SOURce:DATA:TELecom:POINter:MODE BURSt command.
2. To create three pointer adjustments with each burst of pointers, send the SOURce:DATA:TELecom:POINter:NBURst 3 command (if you do not specify the number of pointer adjustments, the instrument assumes you want two pointer adjustments for each burst).
3. Send the SOURce:DATA:TELecom:POINter:ACTion command to initiate a burst of pointer adjustments.

To create continuous pointer adjustments that alternate between up and down, follow these steps:

1. Send the SOURce:DATA:TELecom:POINter:DIRection ALTernate command (you can also specify UP or DOWN as the parameter value).
2. Send the SOURce:DATA:TELecom:POINter:MODE CONTinuous command to initiate the continuous pointer adjustments.

Generating a Failure. Generate a loss of frame failure by sending the SOURce:DATA:TELecom:FAILure:TYPE LOFrame command. You can also specify LOSignal or LOPointer as the parameter value to generate a loss of signal or loss of pointer.

Generating an Alarm. Generate a path alarm indication by sending the SOURce:DATA:TELecom:ALARm PAIS command. You can also specify a variety of other alarms.

NOTE. *You must send the SOURce:DATA:TELecom:FAILure:TYPE NONE command before you generate any alarm. Failures and alarms cannot be generated simultaneously.*

Inserting Errors. Follow these steps to insert errors:

1. Enable error insertion by sending the SOURce:DATA:TELecom:ERRor:ENABle ON command.

2. Insert a B1 section code violation by sending the SOURce:DATA:TELecom:ERRor:TYPE SCV command. You can also specify LCV, PCV, PFEBE, or DATA as the parameter value to insert a B2 line code violation, B3 active path code violation, path far end block error, or payload data bit error, respectively.
3. You can insert errors at a continuous rate or immediately, upon command:
   - Insert continuous errors at a rate of 1E–5 by sending the SOURce:DATA:TELecom:ERRor:RATE 1E–5 command. You can select a variety of error rates depending on the signal rate and error type.
   - Force an immediate error insertion by sending the SOURce:DATA:TELecom:ERRor:IMMediate command.

Creating a Line Frequency Offset. Follow these steps to create a line frequency offset:

1. Select frequency offset pointer adjustments by sending the SOURce:DATA:TELecom:POINter:MODE FOFFset command.
2. Set the clock source to the internal clock by sending the SOURce:CLOCk:SOURce INTernal command.
3. Select line offset with no pointer adjustments by sending the SOURce:CLOCk:OFFSet:MODE LOFFset command.
4. Set the line offset value to –55.1 ppm by sending the SOURce:CLOCk:OFFSet:LVALue –55.1 command. The payload clock offset value will automatically be set to –55.1.

Receiving Signals

This section shows you how to set up the CTS 710 to receive a signal, how to check the status of the signal, and how to drop and view overhead data.

Receiving a Signal Using a Manual Setup

The following example shows you how to manually set up your instrument to receive a normal electrical signal at an STS-1 rate on channel one:

1. Reset the instrument to a known state by sending the *RST command. *Appendix D* lists the default parameter values.
2. Set up the signal physical characteristics by sending the following commands:
   INPUT1:TELecom:RATE STS1
   INPUT1:TELecom:TYPE ELECtrical
   INPUT1:TELecom:LEVel LOW

3. Set the source of the signal by sending the SENSe:DATA:TELecom:SOURce INPUT1 command.

4. Receive the signal through channel one by sending the SENSe:DATA:TELecom:CHANnel 1 command.

Receiving a Signal Using Autoscan

If you do not know what kind of signal you are going to receive, send the SENSe:DATA:TELecom:AUTOscan command. The instrument scans the incoming signal and sets up the receiver to the proper signal rate and payload mapping. The OPC bit is set when autoscan completes execution. To check if the autoscan was able to set up the receiver properly, first send the*ESR? query, and then send the SYSTem:ERRor? query. You will see an error number and description. If the autoscan failed, check to see if a signal is connected.

Checking the Physical Status of a Signal

The following steps show you how to check the physical status of a signal:

1. To check if you are receiving the signal, send the INPUT1:TELecom:STATus? query. If you get a response of LOSIGNAL, try another setup to receive the signal (refer to the *Receiving a Signal Using a Manual Setup* or *Receiving a Signal Using Autoscan* descriptions in this section). If you get a response of MONITOR, you are receiving a monitor signal of low amplitude.

2. Check the optical signal level of the incoming signal by sending the INPUT1:TELecom:OPWR? query. Values for a valid signal are –32 dBm to –5 dBm.

3. If you still are unable to find the proper settings for the incoming signal, check to see if the level is set properly by sending the INPUT1:TELecom:LEVel? query.

Checking Signal Status

To check the status of the received signal, send the SENSe:DATA:TELecom:STATus? query. A response of 8192 indicates a pattern lock on the received signal with no alarms or failures detected. Refer to the description for this query on page 2–140 for list of possible responses. Figure 2–6 on page 2–10 describes how to interpret this type of response.

Dropping the Overhead

Use one of the following ways to drop the overhead to an external protocol analyzer:

- Drop the Section DCC overhead (bytes D1 and D2) to an external protocol analyzer by sending the SENSe:DATA:TELecom:OVERhead:DROP SDCC command. You can also drop Line DCC (bytes D4 through D12) or the F1 byte by using LDCC or F1 as the parameter value.

- Drop the F2 byte to an external protocol analyzer by sending the SENSe:DATA:TELecom:POVerhead:DROP F2 command.

Viewing the Overhead

Follow these steps to freeze the overhead and to query specific overhead bytes:

1. Start acquiring the overhead by sending the INITiate command.
2. To cause a trigger and stop acquiring overhead, send the TRIGger:IMMediate command.
3. Determine the value of the A1 overhead byte on channel one by sending the SENSe:DATA:TELecom:OVERhead DATA? 1,A1,0 query. The table included with the query description on page 2–155 lists the bytes available for selection.
4. To start acquiring overhead again, send the INITiate command.

SONET Signal Testing

This section shows you how to set up several types of tests, how to run these tests, and then how to view the results. The *Storage of Test Results* section starting on page 1–10 describes how test results are stored in the instrument.

Taking BER Measurements

This example shows you how to run a five-minute BER test and view the test results:

1. Connect a cable from the TRANSMIT output to the RECEIVE input.
2. Reset the instrument to a known state by sending the *RST command. *Appendix D* lists the default parameter values.
3. Set the test duration to five minutes by sending the SENSe:DATA:TELecom:TEST:DURation 0,0,5,0 command.
4. Start the test by sending the SENSe:DATA:TELecom:TEST:STARt command.
5. Check whether the test is done by sending the SENSe:DATA:TELecom:TEST:STATus? query. If the first number in the response is 0, the test is done. If the first number is 1, look at the other numbers in the response string to see how long the test has been running. For example, a response of 1,0,0,4,50 indicates that the test has been running for 4 minutes and 50 seconds and is still running. If you had previously set a test duration of 5 minutes, you know that the test will be done in 10 seconds.

6. After the test has completed, send the following queries to view the error measurements:

   - The SENSe:DATA:TELecom:TEST:MEASure:ERRor:ECOUnt:SCV? query returns the number of B1 errors.
   - The SENSe:DATA:TELecom:TEST:MEASure:ERRor:ECOUnt:LCV? query returns the number of B2 errors.
   - The SENSe:DATA:TELecom:TEST:MEASure:ERRor:ECOUnt:PCV? query returns the number of B3 errors.

Measuring Continuous Pointer Adjustments

The following example shows you how to run a test that initiates continuous pointer adjustments, and then view the measurements:

1. Connect a cable from the TRANSMIT output to the RECEIVE input.
2. Reset the instrument to a known state by sending the *RST command. *Appendix D* lists the default parameter values.
3. Set the test duration by sending the SENSe:DATA:TELecom: TEST:DURation 0,0,0,0 command. This test will run continuously.
4. Set up continuous pointer adjustments that alternate between up and down by sending the SENSe:DATA:TELecom:POINter:MODE CONTinuous and SENSe:DATA:TELecom:POINter:DIRection ALTernate commands.
5. Set the pointer adjustment rate to 10 ms by sending the SENSe:DATA: TELecom:POINter:RATE 10 command.
6. Start the test by sending the SENSe:DATA:TELecom:TEST:STARt command.
7. To access the pointer adjustment measurements, send the commands listed below. Keep in mind that while the test is running, these measurements do not represent the final pointer adjustment measurements. Send the SENSe:DATA:TELecom:TEST:STOP command to stop the test, and then the SENSe:DATA:TELecom:MEASure:POINter: queries to access the final pointer adjustment measurements.

   - The SENSe:DATA:TELecom:MEASure:POINter:PPTR? query returns the number of positive (up) pointer adjustments.
   - The SENSe:DATA:TELecom:MEASure:POINter:NPTR? query returns the number of negative (down) pointer adjustments.
   - The SENSe:DATA:TELecom:MEASure:POINter:ICOUNt? query returns the total number of invalid pointers.

Tributary Signal Testing

Add/Drop/Test Option Only

This section discusses five ways to use the Add/Drop/Test Option of your CTS 710 to test the viability of a network element (NE) and portions of the network.

CTS 710 as a Stand-Alone Tributary Test Set

You can use the CTS 710 as a stand-alone tributary test set without using any SONET features. You can verify DS1/DS3 path connection, test path quality, and verify responses to faults. The CTS 710 can both generate and receive DS1/DS3 signals.

The following example generates a DS3 tributary signal and then inserts an alarm:

1. Configure your CTS 710 as shown in Figure 1–7.

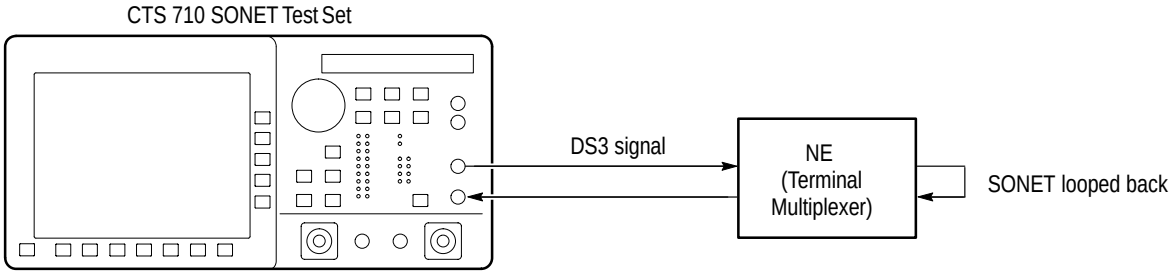


Figure 1–7: Setup for the CTS 710 as a Stand-Alone Tributary Test Set

2. Send the following commands to set up to generate a DS3 tributary signal with a defined framing and pattern:
   SOURce:DATA:TELecom:SOURce OUTPUT3
   SOURce:DATA:TELecom:TRIButary:FRAMing CBIT
   SOURce:DATA:TELecom:TRIButary:PATTern PRBS20
3. Send the following commands to set up to receive the tributary signal:
   SENSe:DATA:TELecom:SOURce INPUT3
   SENSe:DATA:TELecom:TRIButary:FRAMing CBIT
   SENSe:DATA:TELecom:TRIButary:PATTern PRBS20
4. Verify that the CTS 710 is receiving the tributary signal correctly by sending the SENSe:DATA:TELecom:TRIButary:STATus? query. You should receive a response of 8192 indicating a pattern lock on the tributary signal.

5. Now verify the response to faults by inserting an alarm as follows:

   a. Send the SOURce:DATA:TELecom:TRIButary:ALARm AIS command to insert an AIS alarm.

   b. Then send the SENSe:DATA:TELecom:TRIButary:STATus? query. You should receive a response of 64 indicating a DS3 AIS alarm.

Testing the SONET Tributary Payload Mapping

You can use the CTS 710 to create and monitor SONET tributary payloads. This test can verify error events and alarms in the demapped tributary signal, and can measure multiple layer signal quality.

The CTS 710 maps a pattern into the tributary payload of a SONET signal. This signal is then generated by the CTS 710 and transmitted to an NE. The CTS 710 receives the SONET signal and measures both the SONET and mapped tributary signals.

The following example generates a SONET signal with a DS3 tributary mapped into it. Then, after the signal has been transmitted through an NE, the CTS 710 receives the signal and measures the demapped DS3 tributary signal.

1. Configure your CTS 710 as shown in Figure 1–8.

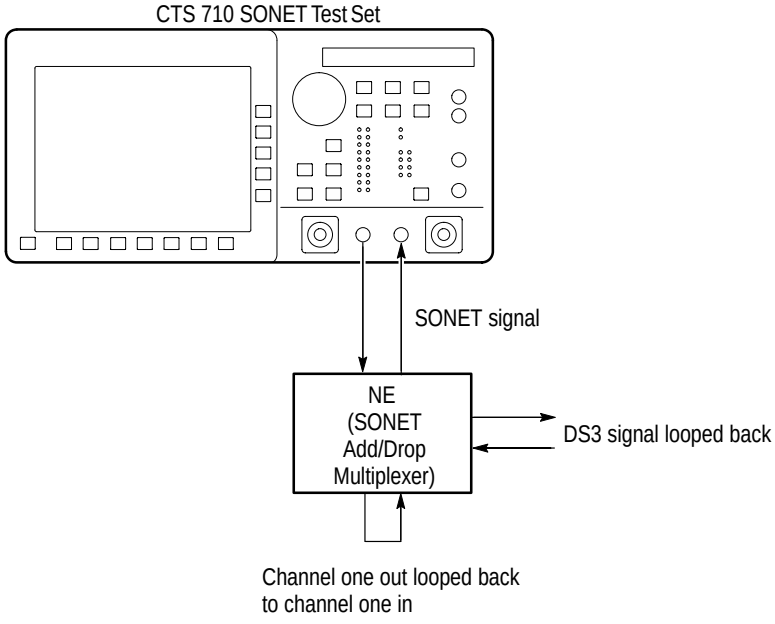


Figure 1–8: Setup for Testing the SONET Tributary Payload Mapping

2. Send the following commands to set up to generate a SONET signal with a defined tributary payload in channel 1:
   SOURce:DATA:TELecom:SOURce OUTPUT1
   OUTPUT1:TELecom:RATE STS1
   SOURce:DATA:TELecom:PAYLoad:MAPPing TRIButary
   SOURce:DATA:TELecom:TRIButary:MAPPing DS3
   SOURce:DATA:TELecom:TRIButary:FRAMing CBIT
   SOURce:DATA:TELecom:TRIButary:PATTern PRBS15
   SOURce:DATA:TELecom:TRIButary:BACKground:PATTern PRBS

3. Send the following commands to set up to receive and demap the tributary signal:
   SENSe:DATA:TELecom:SOURce INPUT1
   INPUT1:TELecom:RATE STS1
   SENSe:DATA:TELecom:PAYLoad:MAPPing TRIButary
   SENSe:DATA:TELecom:TRIButary:MAPPing DS3
   SENSe:DATA:TELecom:TRIButary:FRAMing CBIT
   SENSe:DATA:TELecom:TRIButary:PATTern PRBS15

4. Verify that the CTS 710 is receiving the tributary signal correctly by sending the SENSe:DATA:TELecom:TRIButary:STATus? query. You should receive a response of 8192 indicating a pattern lock on the demapped tributary signal.

Testing the Mapping Capability of a Network Element

You can use the CTS 710 to test the mapping capability of an NE. This test can introduce timing variations in the tributary signal and check for error-free mapping, verify mapping for correct channel assignments, and verify responses to errors, alarms, and failures.

The CTS 710 generates a tributary signal which is sent to an NE. The NE maps this tributary signal into the SONET signal. This signal is received by the CTS 710 which then demaps the tributary signal and measures it.

The following example generates a DS1 tributary signal and then inserts an error:

1. Configure your CTS 710 as shown in Figure 1–9.

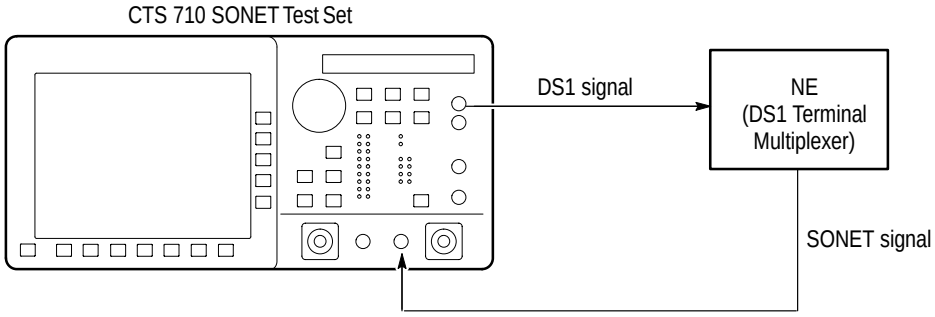


Figure 1–9: Setup for Testing the Mapping Capability of an NE

2. Send the following commands to set up the CTS 710 to generate a DS1 tributary signal with a defined framing:
   SOURce:DATA:TELecom:SOURce OUTPUT2
   SOURce:DATA:TELecom:TRIButary:FRAMing SF
   SOURce:DATA:TELecom:TRIButary:PATTern PRBS20

3. Send the following commands to set up to receive the SONET signal with a tributary signal mapped into it:
   SENSe:DATA:TELecom:SOURce INPUT1
   SENSe:DATA:TELecom:PAYLoad:MAPPing TRIButary
   SENSe:DATA:TELecom:TRIButary:MAPPing VTASYNC
   SENSe:DATA:TELecom:TRIButary:FRAMing SF
   SENSe:DATA:TELecom:TRIButary:PATTern PRBS20

4. Send the following commands to verify the NE response to faults; these commands insert a payload bit error into the tributary signal:
   SOURce:DATA:TELecom:ERRor:TYPE TRIButary
   SOURce:DATA:TELecom:TRIButary:ERRor DATA
   SOURce:DATA:TELecom:ERRor:IMMediate

5. Verify that the CTS 710 is receiving and demapping the tributary signal correctly by sending the SENSe:DATA:TELecom:TRIButary:STATus? query. You should receive a response of 8704 indicating a DS1 bit error and pattern lock in the demapped tributary signal.

Testing the Demapping Capability of a Network Element

You can use the CTS 710 to test the demapping capability of an NE. This test can introduce pointer adjustments, test signal quality, verify correct channel assignments, and verify responses to errors, alarms, and failures.

The CTS 710 generates a tributary signal with a known pattern and maps this signal into the SONET signal. The signal is then sent to an NE, which demaps the tributary signal from the SONET signal. The CTS 710 receiver monitors and measures the demapped tributary signal.

The following example maps a DS1 tributary signal into a SONET signal, receives the demapped DS1 signal from the NE, and then initiates pointer adjustments:

1. Configure your CTS 710 as shown in Figure 1–10.

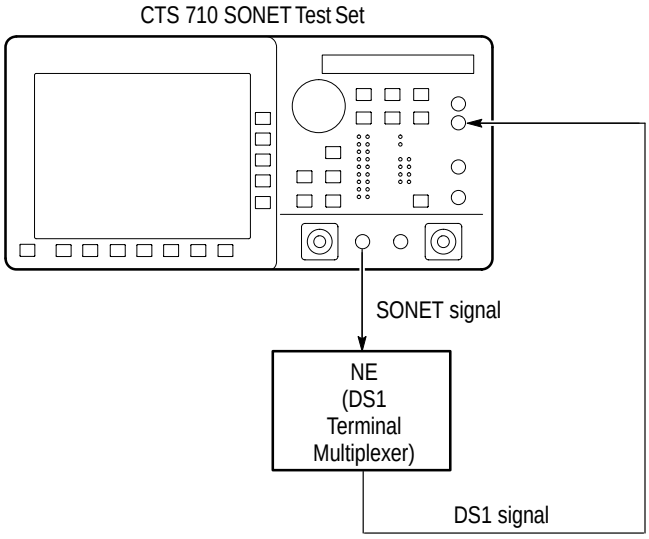


Figure 1–10: Setup for Testing the Demapping Capability of an NE

2. Send the following commands to set up the CTS 710 to map a DS1 tributary signal into channel one of a SONET signal:
SOURce:DATA:TELecom:SOURce OUTPUT1
SOURce:DATA:TELecom:PAYLoad:MAPPing TRIButary
SOURce:DATA:TELecom:TRIButary:CHANnel 1
SOURce:DATA:TELecom:BACKground:PATTern IDLE
SOURce:DATA:TELecom:TRIButary:FRAMing SF
SOURce:DATA:TELecom:TRIButary:PATTern PRBS20
SOURce:DATA:TELecom:TRIButary:MAPPing VTASYNC

3. Send the following commands to set up to receive the demapped tributary signal:
SENSe:DATA:TELecom:SOURce INPUT2
SENSe:DATA:TELecom:TRIButary:FRAMing SF
SENSe:DATA:TELecom:TRIButary:PATTern PRBS20

4. Send the following commands to set up continuous pointer adjustments at a 50 ms rate:
SOURce:DATA:TELecom:POINter:MODE TRIButary
SOURce:DATA:TELecom:TRIButary:POINter:MODE CONTinuous
SOURce:DATA:TELecom:TRIButary:POINter:DIRection ALTernate
SOURce:DATA:TELecom:TRIButary:POINter:RATE 50

5. Verify that the CTS 710 is receiving the demapped tributary signal correctly by sending the SENSe:DATA:TELecom:TRIButary:STATus? query. You should receive a response of 8192 indicating a pattern lock and no errors.

Testing the External Connection of an Add/Drop/Test Set

You can use the CTS 710 to add an external tributary signal into the SONET signal. The testing is controlled by the external tributary test set that generates the tributary signal.

The external tributary test set generates a nonstandard tributary signal. The CTS 710 receives this tributary signal and maps it directly into the SONET signal. The NE receives the SONET signal and demaps the tributary signal. The external tributary test set verifies that the NE demapped the tributary signal correctly.

The following example receives and maps an external DS3 tributary signal into a SONET signal:

1. Configure your CTS 710 as shown in Figure 1–11.

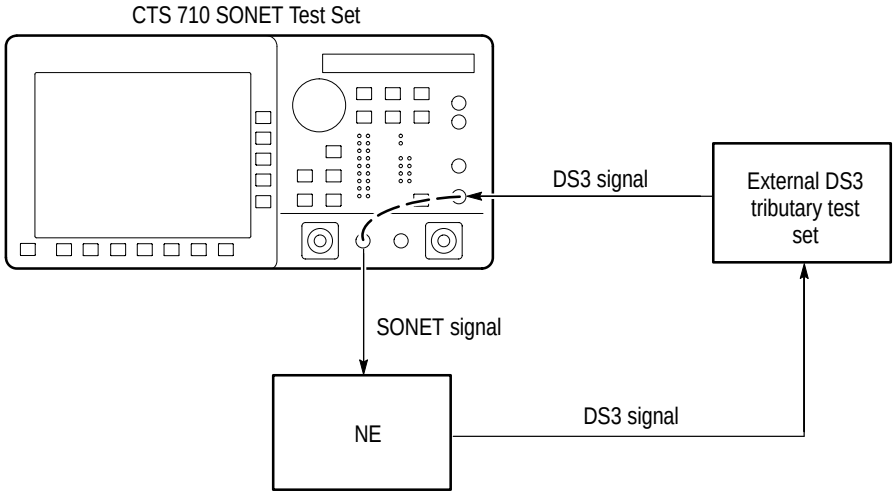


Figure 1–11: Setup for Testing the External Connection of an Add/Drop/Test Set

2. Send the following commands to set up the CTS 710 to map the tributary signal directly into a SONET signal, and then generate the SONET signal (the tributary signal is not measured):
   SOURce:DATA:TELecom:SOURce OUTPUT1
   SOURce:DATA:TELecom:PAYLoad:MAPPing TRIButary
   SOURce:DATA:TELecom:TRIButary:ADD ON
   SOURce:DATA:TELecom:TRIButary:MAPPing DS3

3. The external tributary test set should verify that the NE demapped the tributary signal correctly.

Syntax

This section contains information on the Standard Commands for Programmable Instruments (SCPI) and IEEE 488.2 Common Commands you can use to program your CTS 710.

SCPI Commands and Queries

SCPI is a standard that provides guidelines for remote programming of instruments. These guidelines provide a consistent programming environment for instrument control and data usage. This environment uses defined programming messages, instrument responses, and data format across all SCPI instruments, regardless of manufacturer. The CTS 710 uses a command language derived from this SCPI standard.

The SCPI language is based on a hierarchical tree structure (see Figure 2–1) that represents a subsystem. The top level of the tree is the root node; it is followed by one or more lower-level nodes.

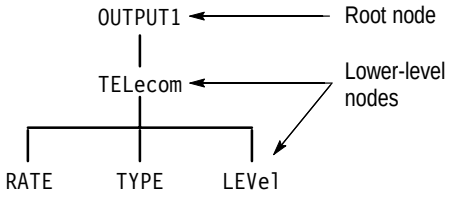


Figure 2–1: Example of SCPI Subsystem Hierarchy Tree

You can create commands and queries from these subsystem hierarchy trees. Commands specify actions for the instrument to perform. Queries return information about the state of the instrument.

Creating Commands

SCPI commands are created by stringing together the nodes of a subsystem hierarchy tree and separating each node by a colon.

In Figure 2–1, OUTPUT1 is the root node and TELecom, RATE, TYPE, and LEVel are the lower-level nodes. To create a SCPI command, start with the root node OUTPUT1 and move down the tree structure adding nodes until you reach the end of a branch. Most commands and some queries have parameters; you must include a value for these parameters. If you specify a parameter value that is out of range, the parameter will be set to the default. The commands sections starting on page 2–11 list the valid values for all parameters.

OUTPUT1:TELecom:LEVel HIGH is an example of a valid SCPI command using the hierarchy tree in Figure 2–1.

Creating Queries

To create a query, start at the root node of a tree structure, move down to the end of a branch, and then add a question mark. OUTPUT1:TELecom:LEVel? is an example of a valid SCPI query using the hierarchy tree in Figure 2–1.

Parameter Types

Parameter types are given for every parameter in the command and query descriptions. The parameters are enclosed in brackets, for example, <pattern>. The parameter type is listed after the parameter and is enclosed in parentheses, for example, (discrete). Some parameter types are defined specifically for the CTS 710 command set and some are defined by ANSI/IEEE 488.2-1987 (see Table 2–1).

Table 2–1: Parameter Types Used in Syntax Descriptions

Parameter Type	Description	Example
binary	Binary numbers	#B0110
binary block[1]	A specified length of binary data	#512234xxxxx . . . where 5 indicates that the following 5 digits (12234) specify the length of the data in bits; xxxxx ... indicates the binary data
boolean	Boolean numbers or values	ON or 1 OFF or 0
discrete	A list of specific values	HIGH, LOW, MID, PRBS23
hexadecimal[2]	Hexadecimal numbers (0–9, A, B, C, D, E, F)	#HAA, #H1
NR1-numeric[2,3]	Integers	0, 1, 15, –1
NR2-numeric[2]	Decimal numbers	1.2, 3.141516, –6.5
NR3-numeric[2]	Floating point numbers	3.1415E–9, –16.1E5
string[4]	Alphanumeric characters (must be within quotation marks)	“Testing 1, 2, 3”

1 **Defined in ANSI/IEEE 488.2 as “Definite Length Arbitrary Block Response Data.”**

2 **An ANSI/IEEE 488.2–1987-defined parameter type.**

3 **Some commands and queries will accept a hexadecimal value even though the parameter type is defined as NR1-numeric.**

4 **Defined in ANSI/IEEE 488.2 as “String Response Data.”**

Abbreviating Commands, Queries, and Parameters

You can abbreviate most SCPI commands, queries, and parameters to an accepted short form. This manual shows these short forms as a combination of upper and lower case letters. The upper case letters tell you what the accepted short form should consist of. As shown in Figure 2–2, you can create a short form by using only the upper case letters. The accepted short form and long form are equivalent and request the same action of the instrument.

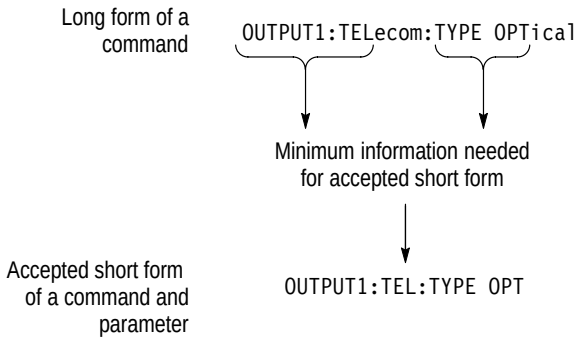


Figure 2–2: Example of Abbreviating a Command

NOTE*. The numeric part of a command or query must always be included in the accepted short form. In Figure 2–2, the "1" of "OUTPUT1" is always included in the command or query.*

Controlling Responses to Queries

You can control the form of responses returned by queries by changing the parameter values of SYSTem:HEADers and SYSTem:VERBose. These two commands control whether the query nodes are returned with the response, and, if the query nodes are returned, whether they are in the long or short form. SYSTem:HEADers controls the presence of the query nodes, and SYSTem:VERBose controls the length of these nodes. Table 2–2 shows the possible combinations of these commands and an example of a query response.

Table 2–2: Using Commands to Control the Response to a Query

SYSTem:HEADers set to:	SYSTem:VERBose set to:	Example of a Response
1 or ON	1 or ON	OUTPUT1:TELECOM:TYPE OPTICAL
1 or ON	0 or OFF	OUTPUT1:TEL:TYPE OPT
0 or OFF	0 or OFF	OPT
0 or OFF	1 or ON	OPTICAL

Chaining Commands and Queries

You can chain several commands or queries together into a single message. To create a chained message, first create a command or query, add a semicolon (;), and then add more commands or queries and semicolons until you are done. Figure 2–3 illustrates a chained message consisting of several commands and queries. The single chained message should end in a command or query, not a semicolon. Responses to any queries in your message are separated by semicolons.

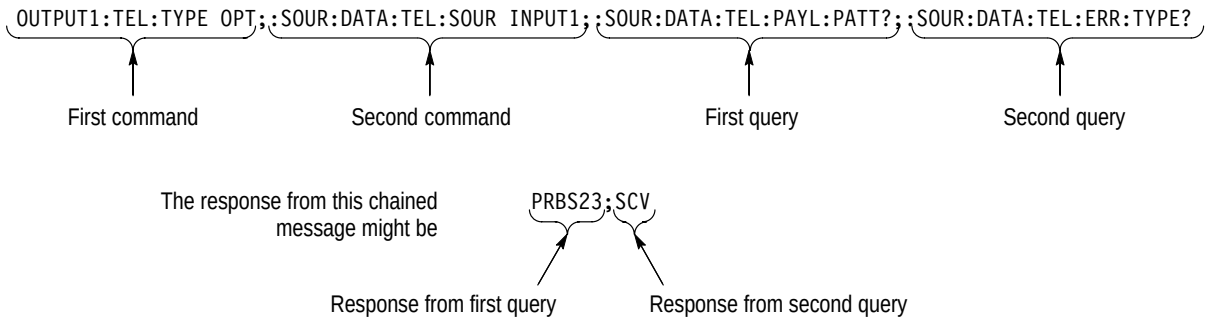


Figure 2–3: Example of Chaining Commands and Queries

If a command or query has the same root and lower-level nodes as the previous command or query, you can omit these nodes. In Figure 2–4, the second command has the same root and lower-level nodes (SOURce:DATA:TELecom) as the first command, so these nodes can be omitted.

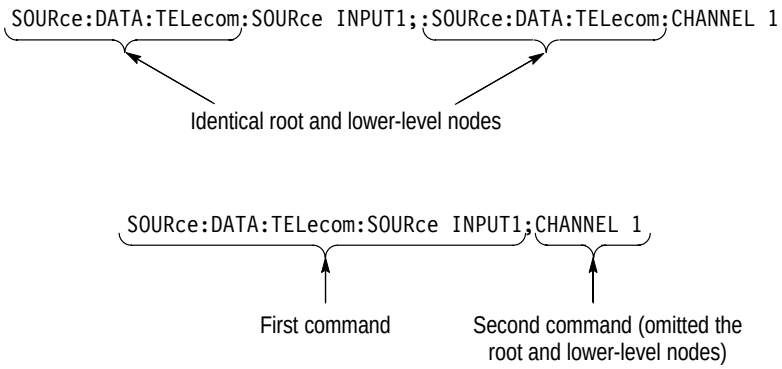


Figure 2–4: Example of Omitting Root and Lower-level Nodes in Chained Message

General Rules

Here are some general rules for using SCPI commands, queries, and parameters:

- You can use single (‘ ’) or double (“ ”) quotation marks for quoted strings, but you cannot use both types of quotation marks for the same string.

  correct: “This string uses quotation marks correctly.”

  correct: ‘This string also uses quotation marks correctly.’

  incorrect: “This string does not use quotation marks correctly.’

- You can use upper case, lower case, or a mixture of both cases for all commands, queries, and parameters.

  ```
  INPUT1:TELECOM:TYPE ELECTRICAL
  ```

  is the same as

  ```
  input1:telecom:type electrical
  ```

  and

  ```
  INPUT1:telecom:Type ELECTRICAL
  ```

- No embedded spaces are allowed between or within nodes.

  correct: `OUTPUT1:TELecom:TYPE OPTical`

  incorrect: `OUTPUT1: TELecom: TYPE OPTical`

  incorrect: `OU TPUT1:TELe com:TYPE OPTical`

IEEE 488.2 Common Commands

Description

ANSI/IEEE Standard 488.2 defines the codes, formats, protocols, and usage of common commands and queries used on the GPIB interface between the controller and the instruments. The CTS 710 complies with this standard.

Command and Query Structure

The syntax for an IEEE 488.2 common command is an asterisk (*) followed by a command and, optionally, a space and parameter value. The syntax for an IEEE 488.2 common query is an asterisk (*) followed by a query and a question mark. All of the common commands and queries are listed in the last part of the *Syntax and Commands* section. The following are examples of common commands:

- `*ESE 16`
- `*CLS`

The following are examples of common queries:

- `*ESR?`
- `*IDN?`

Functional Command Groups

All of the commands and queries in *Syntax and Commands* are organized into functional groups. Each section covers one functional group. For example, *Transmit Commands* contains all commands and queries that allow you to set up and transmit a signal. The commands and queries within each functional group are further organized into subsystems. For example, within *Transmit Commands* are subsystems that allow you to set the physical characteristics of a signal (the OUTPUT1 subsystem) and pointer adjustments (the SOURce:DATA:TELecom: POINter subsystem).

The functional groups and their subsystems are shown in Table 2–3.

Table 2–3: Functional Groups and their Subsystems

Functional Group	Subsystem	Description	Starts on Page
Transmit Commands	OUTPUT1	Controls physical setup of transmitted SONET signal	2–12
	OUTPUT2	Sets the characteristics of the transmitted or dropped DS1 tributary signal (Add/Drop/Test Option Only)	2–16
	OUTPUT3	Sets the characteristics of the transmitted or dropped DS3 tributary signal (Add/Drop/Test Option Only)	2–19
	SOURce:CLOCk	Controls transmitter clock	2–21
	SOURce:DATA:TELecom	Controls transmitter setup	2–28
	SOURce:DATA:TELecom:OVERhead and POVerhead	Controls transmitter overheads	2–37
	SOURce:DATA:TELecom:ERRor, ALARm, and FAILure	Controls transmitter abnormalities	2–52
	SOURce:DATA:TELecom:POINter	Controls transmitter pointers	2–61
	SOURce:DATA:TELecom:TRIButary	Controls transmitted or dropped tributary signal (Add/Drop/Test Option Only)	2–83
	SOURce:DATA:TELecom:TRIButary: ERRor, ALARm, and FAILure	Controls abnormal conditions in the transmitted or dropped tributary signal (Add/Drop/Test Option Only)	2–97
	SOURce:DATA:TELecom:TRIButary: POINter	Controls pointers in the transmitted or dropped tributary signal (Add/Drop/Test Option Only)	2–102
	SOURce:DATA:TELecom:TRIButary: POVerhead	Controls path overhead in the transmitted or dropped tributary signal (Add/Drop/Test Option Only)	2–95

Table 2–3: Functional Groups and their Subsystems (Cont.)

Functional Group	Subsystem	Description	Starts on Page
Receive Commands	INPUT1	Sets up physical connection of received SONET signal	2–124
	INPUT2	Sets the characteristics of the received or added DS1 tributary signal (Add/Drop/Test Option Only)	2–130
	INPUT3	Sets the characteristics of the received or added DS3 tributary signal (Add/Drop/Test Option Only)	2–134
	SENSe:DATA:TELecom	Sets up receiver	2–138
	SENSe:DATA:TELecom:TEST	Starts and stops measurements	2–148
	SENSe:DATA:TELecom:OVERhead and POVerhead	Allows access to receiver overheads	2–155
	SENSe:DATA:TELecom:MEASure	Allows access to measurements	2–162
	SENSe:DATA:TELecom:MEASure:STESts	Controls pass/fail tests	2–180
	SENSe:DATA:TELecom:AUTOscan	Automatically sets up receiver	2–192
	SENSe:DATA:TELecom:TRIButary	Controls viewing of tributary signal (Add/Drop/Test Option Only)	2–193
	SENSe:DATA:TELecom:TRIButary:POVerhead	Controls path overhead in the received or dropped tributary signal (Add/Drop/Test Option Only)	2–158
	SENSe:DATA:TELecom:MEASure:TRIButary	Allows access to measurements of tributary signals (Add/Drop/Test Option Only)	2–210
Transmitter and Receiver Setup Commands	INSTrument	Controls transmitter and receiver settings	2–221
Trigger and Capture Commands	INITiate	Starts overhead capture	2–225
	TRIGger:IMMediate	Stops overhead capture	2–225
Input/Output Commands	MMEMory	Controls disk drive	2–227
	DISPlay	Sets display intensity	2–234
	HCOPy	Controls printer	2–236
	SYSTem:COMMunicate	Controls RS-232 setups	2–241
Instrument Control Commands	SYSTem	Controls general instrument functions	2–253
Diagnostic Commands	DIAGnostic	Controls self tests	2–271
IEEE 488.2 Common Commands	(no subsystems; every command and query begins with *)	Allows access to generic commands	2–279

Each functional group section begins with a description of the functional group and is followed by a list of the subsystems included in the functional group. Then, for each of the subsystems, a description and hierarchy tree are given.

Each command and query within each subsystem are listed in the functional group sections in the format illustrated in Figure 2–5. For the sake of clarity, two tables are always given even though the parameters may be identical.

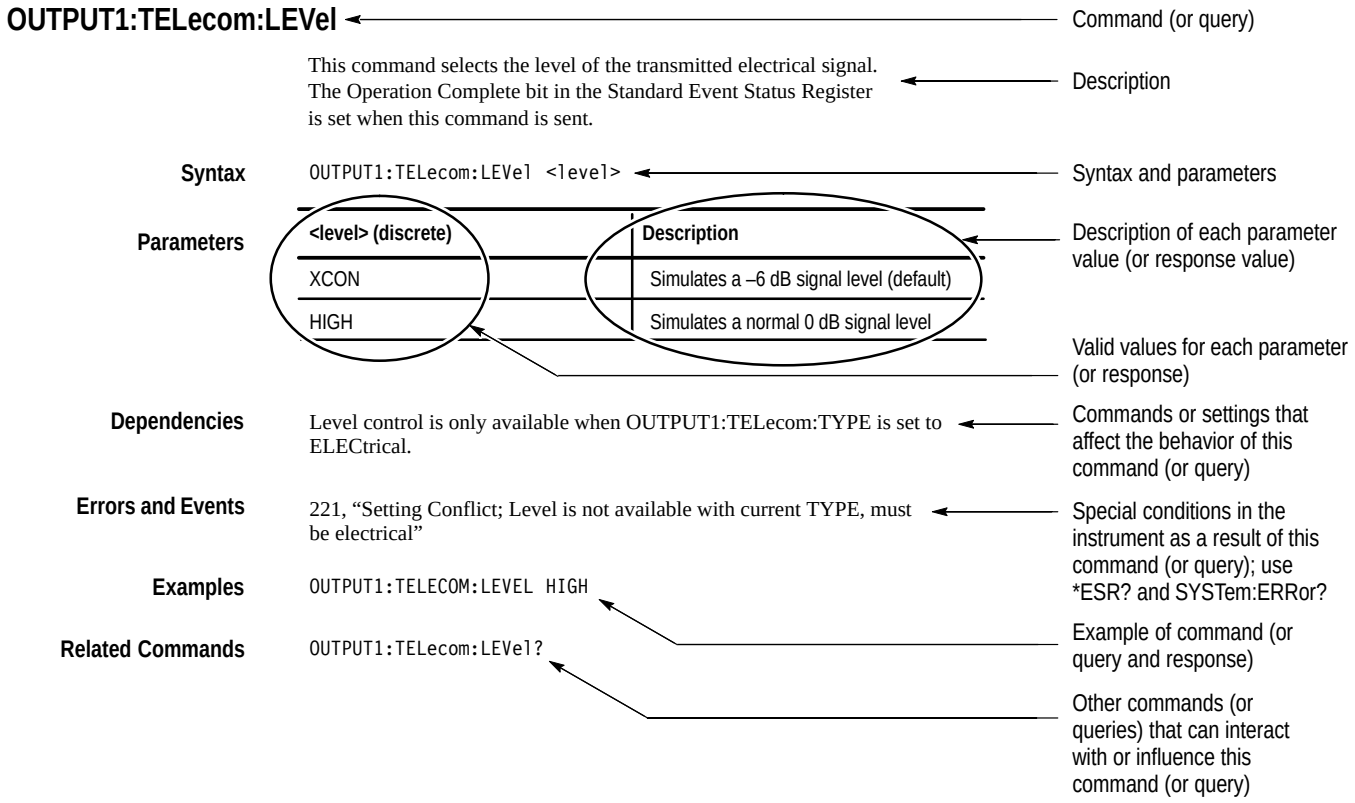


Figure 2–5: Example of Command Listing

***NOTE**. Some commands and queries follow a different format than shown in Figure 2–5 (for example, the* SENSe:DATA:TELecom:MEASure *queries). An explanation of this format is found at the beginning of the section containing the commands and queries.*

In the Syntax and Commands section you will see a different kind of Parameter or Response table for a few commands or queries. Figure 2–6 shows you an example of a <decimal value> response table. The parameter or response value returned is the sum of the decimal values listed in the left column and depends on which bits are set. Follow the step numbers in the example to interpret a <decimal value> parameter or response.

Table X–X: Response Table

<decimal value> (NR1-numeric)	bit	definition
1	0	LOF
2	1	LOF
4	2	OOF
8	3	LOP
16	4	Line AIS
32	5	Path AIS
64	6	Error
128	7	Undefined
256	8	K1/K2 change
512	9	Line FERF
1024	10	Path FERF
2048	11	Pointer adjust
4096	12	NDF
8192	13	Pattern lock
16384	14	Not used
32768	15	Not used

1 A response of 9216 is received.

2 Find which decimal values add up to the response of 9216 (1024 + 8192 = 9216).

3 Read across the selected decimal values to the bit and definition columns to interpret the response. In this example, bits 10 and 13 are set indicating a path yellow and pattern lock.

Figure 2–6: How to Interpret a <decimal value> Parameter or Response

Transmit Commands

The Transmit Commands allow you to set the conditions for the signal to be transmitted, including abnormal conditions. This section contains all of the commands and queries for each of the following Transmit subsystems:

- OUTPUT1
- OUTPUT2 (Add/Drop/Test Option Only)
- OUTPUT3 (Add/Drop/Test Option Only)
- SOURce:CLOCk
- SOURce:DATA:TELecom
- SOURce:DATA:TELecom:OVERhead and POVerhead
- SOURce:DATA:TELecom:ERRor, ALARm, and FAILure
- SOURce:DATA:TELecom:POINter
- SOURce:DATA:TELecom:TRIButary (Add/Drop/Test Option Only)
- SOURce:DATA:TELecom:TRIButary:ERRor, ALARm, and FAILure (Add/Drop/Test Option Only)
- SOURce:DATA:TELecom:TRIButary:POINter (Add/Drop/Test Option Only)
- SOURce:DATA:TELecom:TRIButary:POVerhead (Add/Drop/Test Option Only)

OUTPUT1 Subsystem

This section describes the commands and queries that set the rate, type, and level of the signal to be transmitted. Figure 2–7 shows the hierarchy tree for this subsystem.

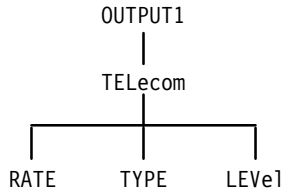


Figure 2–7: OUTPUT1 Subsystem

OUTPUT1:TELecom:RATE

This command selects the output rate of the signal.

Syntax

```
OUTPUT1:TELecom:RATE <rate>
```

Parameters

<rate> (discrete)	description
STS1	51.84 MHz (default)
STS3	155.52 MHz
STS12	622.08 MHz (requires the optical option)

Dependencies

None

Errors and Events

221, “Settings conflict; Rate is not available with current Line Interface module or operating mode”

221, “Settings conflict; Optical module required”

Examples

```
OUTPUT1:TELECOM:RATE STS1
```

Related Commands

OUTPUT1:TELecom:TYPE
SOURce:DATA:TELecom:SOURce

OUTPUT1:TELecom:RATE?

This query returns the current setting of the transmitter rate.

Syntax `OUTPUT1:TELecom:RATE?`

Response

<rate> (discrete)	description
STS1	51.84 MHz (default)
STS3	155.52 MHz
STS12	622.08 MHz (requires the optical option)

Dependencies None

Errors and Events None

Examples Query: `OUTPUT1:TELECOM:RATE?`

Response: `STS1`

Related Commands OUTPUT1:TELecom:RATE

OUTPUT1:TELecom:TYPE

This command selects the output signal type.

Syntax `OUTPUT1:TELecom:TYPE <type>`

Parameters

<type> (discrete)	description
ELECtrical	Electrical signal output (default)
OPTical	Optical signal output (requires the optical option)

Dependencies None

Errors and Events 221, “Settings conflict; Type is not available with current Line Interface module”

Examples `OUTPUT1:TELECOM:TYPE ELECTRICAL`

Related Commands None

OUTPUT1:TELecom:TYPE?

This query returns the current setting of the transmitter type.

Syntax `OUTPUT1:TELecom:TYPE?`

Response

<type> (discrete)	description
ELECtrical	Electrical signal output (default)
OPTical	Optical signal output

Dependencies None

Errors and Events None

Examples Query: `OUTPUT1:TELECOM:TYPE?`

Response: `OPTICAL`

Related Commands OUTPUT1:TELecom:TYPE
OUTPUT1:TELecom:RATE

OUTPUT1:TELecom:LEVel

This command selects the level of the transmitted electrical signal.

Syntax `OUTPUT1:TELecom:LEVel <level>`

Parameters

<level> (discrete)	description
XCONnect	Simulates a –6 dB signal level (default)
HIGH	Simulates a normal 0 dB signal level

Dependencies This command applies only when OUTPUT1:TELecom:TYPE is set to ELECtrical.

Errors and Events 221, “Settings conflict; Argument is not valid in current instrument state”

Examples

```
OUTPUT1:TELECOM:LEVEL HIGH
```

Related Commands None

OUTPUT1:TELecom:LEVel?

This query returns the current setting of the transmitter level.

Syntax

```
OUTPUT1:TELecom:LEVel?
```

Response

<level> (discrete)	description
XCONnect	Simulates a –6 dB signal level
HIGH	Simulates a normal 0 dB signal level

Dependencies None

Errors and Events None

Examples Query:

```
OUTPUT1:TELECOM:LEVEL?
```

Response:

```
HIGH
```

Related Commands OUTPUT1:TELecom:LEVel

OUTPUT2 Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that set the characteristics of the transmitted or dropped DS1 tributary signal.

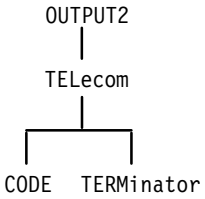


Figure 2–8: OUTPUT2 Subsystem

OUTPUT2:TELecom:CODE

Add/Drop/Test Option Only

This command selects the signal coding of the DS1 tributary signal.

Syntax

```
OUTPUT2:TELecom:CODE <trib1 output code>
```

Parameters

<trib1 output code> (discrete)	description
AMI	Alternate Mark Inversion (default)
B8ZS	Bipolar 8 Zero Substitution

Dependencies

The instrument must be transmitting or dropping a DS1 tributary signal for this command to apply.

Errors and Events

None

Examples

```
OUTPUT2:TELECOM:CODE AMI
```

Related Commands

SOURce:DATA:TELecom:SOURce
SENSe:DATA:TELecom:TRIButary:DROP

OUTPUT2:TELecom:CODE?

Add/Drop/Test Option Only

This query returns the current setting of the signal coding for the DS1 tributary signal.

Syntax `OUTPUT2:TELecom:CODE?`

Response

<trib1 output code> (discrete)	description
AMI	Alternate Mark Inversion (default)
B8ZS	Bipolar 8 Zero Substitution

Dependencies None

Errors and Events None

Examples Query: `OUTPUT2:TELECOM:CODE?`

Response: `AMI`

Related Commands OUTPUT2:TELecom:CODE

OUTPUT2:TELecom:TERMinator

Add/Drop/Test Option Only

This command selects the signal terminator for the DS1 transmit connector.

Syntax `OUTPUT2:TELecom:TERMinator <trib1 output termin>`

Parameters

<trib1 output termin> (discrete)	description
BALanced	120 Ω connector (default)

Dependencies The instrument must be transmitting or dropping a DS1 tributary signal for this command to apply.

Errors and Events None

Examples `OUTPUT2:TELECOM:TERMINATOR BALANCED`

Related Commands SOURce:DATA:TELecom:SOURce
SENSe:DATA:TELecom:TRIButary:DROP

OUTPUT2:TELecom:TERMinator?

Add/Drop/Test Option Only

This returns the current setting of the DS1 transmit connector signal terminator.

Syntax `OUTPUT2:TELecom:TERMinator?`

Response

<trib1 output termin> (discrete)	description
BALanced	120 Ω connector (default)

Dependencies None

Errors and Events None

Examples Query: `OUTPUT2:TELECOM:TERMINATOR?`

Response: `BALANCED`

Related Commands OUTPUT2:TELecom:TERMinator

OUTPUT3 Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that set the characteristics of the transmitted or dropped DS3 tributary signal.

Figure 2–9: OUTPUT3 Subsystem

OUTPUT3:TELecom:RATE

Add/Drop/Test Option Only

This command selects the DS3 tributary output rate.

Syntax `OUTPUT3:TELecom:RATE <trib2 output rate>`

Parameters

<trib2 output rate> (discrete)	description
DS3	44.736 Mb/s (default)

Dependencies None

Errors and Events None

Examples `OUTPUT3:TELECOM:RATE DS3`

Related Commands SOURce:DATA:TELecom:SOURce

OUTPUT3:TELecom:RATE?

Add/Drop/Test Option Only

This query returns the current setting of the DS3 tributary output rate.

Syntax `OUTPUT3:TELecom:RATE?`

Response

<trib2 output rate> (discrete)	description
DS3	44.736 Mb/s (default)

Dependencies None

Errors and Events None

Examples Query: `OUTPUT3:TELECOM:RATE?`

Response: `DS3`

Related Commands OUTPUT3:TELecom:RATE

SOURce:CLOCk Subsystem

This section describes the commands and queries that control the transmitter clock. You must set the SOURce:DATA:TELecom:POINter:MODE command to FOFFset for any of the SOURce:CLOCk subsystem commands to be valid. Table 2–4 shows the interaction between the major commands of this subsystem. Refer to this table to see which combinations of commands and parameters are valid. Figure 2–10 shows the hierarchy tree for this subsystem.

Table 2–4: Interaction Between SOURce:CLOCk Commands

To control:	Set SOURce:CLOCk: SOURce to:	Set SOURce:CLOCk: OFFSet:MODE to:	Set SOURce:CLOCk: OFFSet:LVALue to:	Set SOURce:CLOCk: OFFSet:PVALue to:
Pointer movements	INTernal, BITS, or RECovered	POINters	Set to 0; no changes allowed	Any value from –100 ppm to +100 ppm in increments of 0.1 ppm
Line offset, no pointers	INTernal or RECovered	LOFFset	SONET rates: –100 ppm to +100 ppm DS1 rate: –130 ppm to +130 ppm DS3 rate: –130 ppm to +130 ppm All ranges in increments of 0.1 ppm	Automatically set to the same value as LVALue; you can not directly change PVALue

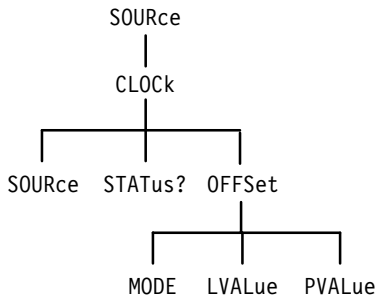


Figure 2–10: SOURce:CLOCk Subsystem

SOURce:CLOCk:SOURce

This command selects the clock for the transmitter. The OPC bit in the Standard Event Status Register is set when this command has completed execution.

Syntax `SOURce:CLOCk:SOURce <clock source>`

Parameters

<clock source> (discrete)	description
INTernal	Internal clock (default)
BITs	External BITS clock
RECovered	Recovered from the received signal
TEXTernal	Tributary external clock (Add/Drop/Test Option Only)

Dependencies RECovered is not allowed if you are set up to receive a tributary signal at the same time you are set up to transmit a SONET signal.

Errors and Events None

Examples

```
SOURCE:CLOCK:SOURCE INTERNAL
```

Related Commands None

SOURce:CLOCk:SOURce?

This query returns the selected clock for the transmitter.

Syntax

```
SOURce:CLOCk:SOURce?
```

Response

<clock source> (discrete)	description
INTernal	Internal clock (default)
BITs	External BITS clock
RECovered	Recovered from the received signal
TEXTernal	Tributary external clock (Add/Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:CLOCK:SOURCE?`

Response: `INTERNAL`

Related Commands

SOURce:CLOCk:SOURce

SOURce:CLOCk:STATus?

This query returns the status of the clock phase locked loop. Use this query to determine if you have a lock on an external clock source or after you change the clock source.

Syntax

`SOURce:CLOCk:STATus?`

Response

<clock status> (boolean)	description
0	Unlocked
1	Locked

Dependencies

None

Errors and Events

None

Examples

Query: `SOURCE:CLOCK:STATUS?`

Response: 1

Related Commands

SOURCe:CLOCk:OFFSet:MODE

SOURce:CLOCk:OFFSet:MODE

This command selects the clock offset mode and determines how the commands SOURce:CLOCk:OFFSet:MODE:LVALue and SOURce:CLOCk:OFFSet:MODE:PVALue interact. When you send the SOURce:CLOCk:OFFSet:MODE command, the values of LVALue and PVALue are reset to 0 which might create a discontinuity in the output signal for a brief time. Then you can change LVALue and PVALue to valid values (Table 2–4 on page 2–21 describes the interaction between the major SOURce:CLOCk commands).

Syntax

```
SOURce:CLOCk:OFFSet:MODE <clock offset mode>
```

Parameters

<clock offset mode> (discrete)	description
LOFFset	Changes to LVALue to control line offset
POINters	Changes to PVALue are allowed

Dependencies

POINters is valid only for SONET rates and when SOURce:CLOCk:SOURce is set to INTernal, BITS, or RECovered.

LOFFset is valid only when SOURce:CLOCk:SOURce is set to INTernal, BITS, or RECovered. LOFFset does not apply when transmitting a tributary signal while using a RECovered clock source.

Errors and Events

None

Examples

```
SOURCE:CLOCK:OFFSET:MODE LOFFSET
```

Related Commands

SOURce:CLOCk:OFFSet:LVALue
SOURce:CLOCk:OFFSet:PVALue

SOURce:CLOCk:OFFSet:MODE?

This query returns the clock offset mode.

Syntax

```
SOURce:CLOCk:OFFSet:MODE?
```

Response

<clock offset mode> (discrete)	description
LOFFset	Changes to LVALue to control line offset
POINters	Changes to PVALue are allowed

Dependencies None

Errors and Events None

Examples Query: `SOURCE:CLOCK:OFFSET:MODE?`

Response: `LOFFSET`

Related Commands SOURce:CLOCk:OFFSet:MODE

SOURce:CLOCk:OFFSet:LVALue

This command selects the line clock offset value in ppm (parts per million).

Syntax `SOURce:CLOCk:OFFSet:LVALue <line clock offset>`

Parameters

<line clock offset> (NR2-numeric)	description
SONET rates: –100 ppm to +100 ppm DS1 rate: –100 ppm to +100 ppm DS3 rate: –100 ppm to +100 ppm All ranges in increments of 0.1 ppm	The line clock offset is set to this value (default = 0)

Dependencies This command applies only when SOURce:CLOCK:SOURce is set to INTernal or RECovered and SOURce:CLOCk:OFFSet:MODE is set to LOFFset.

Errors and Events 221, “Settings conflict; Frequency offset disabled with current transmit clock”

Examples `SOURCE:CLOCK:OFFSET:LVALUE 20`

Related Commands SOURce:CLOCk:OFFSet:MODE

SOURce:CLOCk:OFFSet:LVALue?

This query returns the value of the line clock offset in ppm (parts per million).

Syntax `SOURce:CLOCk:OFFSet:LVALue?`

Response

<line clock offset> (NR2-numeric)	description
SONET rates: –100 ppm to +100 ppm DS1 rate: –100 ppm to +100 ppm DS3 rate: –100 ppm to +100 ppm All ranges in increments of 0.1 ppm	The line clock offset is set to this value (default = 0)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:CLOCK:OFFSET:LVALUE?`

Response: `-10.1`

Related Commands SOURce:CLOCk:OFFSet:LVALue

SOURce:CLOCk:OFFSet:PVALue

This command selects the payload clock offset value in ppm (parts per million).

Syntax `SOURce:CLOCk:OFFSet:PVALue <payload clock offset>`

Parameters

<payload clock offset> (NR2-numeric)	description
Any decimal number in the range –100 ppm to +100 ppm in increments of 0.1 ppm	The payload clock offset is set to this value (default = 0)

Dependencies This command is valid only when SOURce:CLOCk:OFFSet:MODE is set to POINters and when transmitting or receiving a SONET signal.

Errors and Events None

Examples

```
SOURCE:CLOCK:OFFSET:PVALUE 20
```

Related Commands SOURce:CLOCk:OFFSet:MODE

SOURce:CLOCk:OFFSet:PVALue?

This query returns the value of the payload clock offset in ppm (parts per million).

Syntax

```
SOURce:CLOCk:OFFSet:PVALue?
```

Response

<payload clock offset> (NR2-numeric)	description
Any decimal number in the range –100 ppm to +100 ppm in increments of 0.1 ppm	The payload clock offset is set to this value (default = 0)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:CLOCK:OFFSET:PVALUE?`

Response: 0

Related Commands SOURce:CLOCk:OFFSet:LVALue

SOURce:DATA:TELecom Subsystem

This section describes the commands and queries that set up the structure of the signal to be transmitted for both active and inactive channels. Figure 2–11 shows the hierarchy tree for this subsystem.

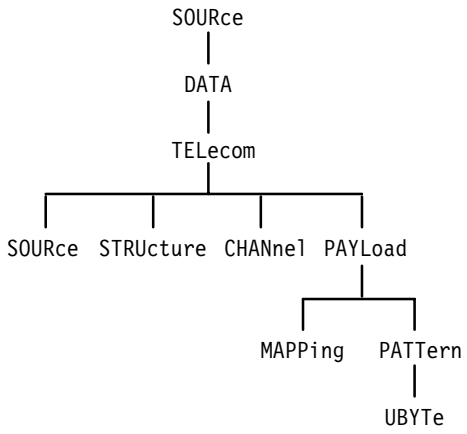


Figure 2–11: SOURce:DATA:TELecom Subsystem

SOURce:DATA:TELecom:SOURce

This command selects the output signal source.

Syntax

```
SOURce:DATA:TELecom:SOURce <source>
```

Parameters

<source> (discrete)	description
OUTPUT1	All SONET rates; normal mode (default)
OUTPUT2	DS1 tributary signal (Add/Drop/Test Option Only)
OUTPUT3	DS3 tributary signal (Add/Drop/Test Option Only)
INPUT1	Through mode

Dependencies

None

Errors and Events

221, “Settings conflict; Not available without tributary option”

Examples `SOURCE:DATA:TELECOM:SOURCE OUTPUT1`

Related Commands OUTPUT1:TELecom:RATE

SOURce:DATA:TELecom:SOURce?

This query returns the current setting of the output signal source.

Syntax `SOURce:DATA:TELecom:SOURce?`

Response

<source> (discrete)	description
OUTPUT1	All SONET rates; normal mode (default)
OUTPUT2	DS1 tributary signal (Add/Drop/Test Option Only)
OUTPUT3	DS3 tributary signal (Add/Drop/Test Option Only)
INPUT1	Through mode

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:SOURCE?`

Response: `OUTPUT1`

Related Commands SOURce:DATA:TELecom:SOURce

SOURce:DATA:TELecom:STRUcture

This command selects the structure of a signal.

Syntax `SOURce:DATA:TELecom:STRUcture <output structure>`

Parameters

<output structure> (discrete)	description
STS1	STS-1 structure (default)
STS3C	STS-3c structure

Dependencies SOURce:DATA:TELecom:STRUcture STS3C is not valid at an STS-1 rate.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:STRUCTURE STS1
```

Related Commands OUTPUT1:TELecom:RATE

SOURce:DATA:TELecom:STRUcture?

This query returns the selected structure.

Syntax

```
SOURce:DATA:TELecom:STRUcture?
```

Response

<output structure> (discrete)	description
STS1	STS-1 structure (default)
STS3C	STS-3c structure

Dependencies None

Errors and Events None

Examples

Query:

```
SOURCE:DATA:TELECOM:STRUCTURE?
```

Response:

```
STS1
```

Related Commands SOURce:DATA:TELecom:STRUcture

SOURce:DATA:TELecom:CHANnel

This command selects the active channel. For example, a SONET STS-3 signal can have three STS-1 signals multiplexed into it. This command allows you to choose which of the three to test.

Syntax

```
SOURce:DATA:TELecom:CHANnel <channel>
```

Parameters

<channel> (NR1-numeric)	description
1	STS-1 rate or STS-3 rate with STS-3c structure (default)
1 to 3	STS-3 rate with STS-1 structure
1 to 4	STS-12 rate with STS-3c structure
1 to 12	STS-12 rate with STS-1 structure

Dependencies

OUTPUT1:TELecom:RATE must be set to STS3 or STS12 if you choose a channel value greater than one.

Errors and Events

221, “Settings conflict; Only one channel is available”

500, “Execution warning; Numeric value greater than maximum limit”

Examples

```
SOURCE:DATA:TELECOM:CHANNEL 1
```

Related Commands

OUTPUT1:TELecom:RATE
SOURce:DATA:TELecom:STRUcture

SOURce:DATA:TELecom:CHANnel?

This query returns the selected active channel.

Syntax

```
SOURce:DATA:TELecom:CHANnel?
```

Parameters

<channel> (NR1-numeric)	description
1	STS-1 rate or STS-3 rate with STS-3c structure (default)
1 to 3	STS-3 rate with STS-1 structure

<channel> (NR1-numeric)	description
1 to 4	STS-12 rate with STS-3c structure
1 to 12	STS-12 rate with STS-1 structure

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:CHANNEL?`

Response: 1

Related Commands SOURce:DATA:TELecom:CHANnel

SOURce:DATA:TELecom:PAYLoad:MAPPing

This command selects the payload mapping. The parameter changes the value in the C2 byte and fills the STS with the pattern selected by the SOURce:DATA:TELecom:PAYLoad:PATTern command. Or, the STS can be filled with a tributary payload.

Syntax `SOURce:DATA:TELecom:PAYLoad:MAPPing <mapping>`

Parameters

<mapping> (discrete)	description
EQUipped	C2 Path Overhead byte is set to 01 (default)
UNEQuipped	C2 Path Overhead byte is set to 00
TRIButary	Allows tributary payload mapping (Add/ Drop/Test Option Only)

Dependencies Select EQUipped or UNEQuipped to use the SOURce:DATA:PAYLoad:PATTern command.

Errors and Events 221, “Settings conflict; Not available without tributary option”

Examples SOURCE:DATA:TELECOM:PAYLOAD:MAPPING EQUIPPED

Related Commands None

SOURce:DATA:TELecom:PAYLoad:MAPPing?

This query returns the current setting of the payload mapping.

Syntax SOURce:DATA:TELecom:PAYLoad:MAPPing?

Response

<mapping> (discrete)	description
EQUipped	C2 Path Overhead byte set to 01 (default)
UNEQuipped	C2 Path Overhead byte set to 00
TRIButary	Allows tributary payload mapping (Add/ Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples Query: SOURCE:DATA:TELECOM:PAYLOAD:MAPPING?

Response: EQUIPPED

Related Commands SOURce:DATA:TELecom:PAYLoad:MAPPing

SOURce:DATA:TELecom:PAYLoad:PATTern

This command selects the test pattern to be placed in the payload of the active channel.

Syntax SOURce:DATA:TELecom:PAYLoad:PATTern <pattern>

Parameters

<pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 is placed in the payload (default)
PRBS9	A pseudo-random binary sequence of length 2^{9}–1 is placed in the payload
PRBS15	A pseudo-random binary sequence of length 2^{15}–1 is placed in the payload
PRBS20	A pseudo-random binary sequence of length 2^{20}–1 is placed in the payload
AZEROs	All zeros are placed in the payload
AONEs	All ones are placed in the payload
UBYTe	A user-defined byte is placed in the payload

Dependencies None

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:PAYLOAD:PATTERN PRBS23
```

Related Commands SOURce:DATA:TELecom:PAYLoad
SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe

SOURce:DATA:TELecom:PAYLoad:PATTern?

This query returns the current test pattern for the active channel payload.

Syntax

```
SOURce:DATA:TELecom:PAYLoad:PATTern?
```

Response

<pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 is in the payload (default)
PRBS9	A pseudo-random binary sequence of length 2^{9}–1 is in the payload
PRBS15	A pseudo-random binary sequence of length 2^{15}–1 is in the payload

(continued on next page)

<pattern> (discrete)	description
PRBS20	A pseudo-random binary sequence of length 2^{20}–1 is in the payload
AZEROs	All zeros are in the payload
AONEs	All ones are in the payload
UBYTe	A user-defined byte is in the payload

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:PAYLOAD:PATTERN?`

Response: `PRBS23`

Related Commands SOURce:DATA:TELecom:PAYLoad:PATTern

SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe

This command selects the internally generated payload fixed pattern.

Syntax `SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe <fixed pattern>`

Parameters

<fixed pattern> (NR1-numeric)[1]	description
Any integer in the range 0 to 255 (hexadecimal 00 to FF)	The payload pattern is set to this value (default = 0)

1 **A hexadecimal value is also acceptable.**

Dependencies SOURce:DATA:TELecom:PAYLoad:PATTern must be set to UBYTE for this command to apply.

Errors and Events None

Examples SOURCE:DATA:TELECOM:PAYLOAD:PATTERN:UBYTE 104

Related Commands SOURce:DATA:TELecom:PAYLoad:PATTern

SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe?

This query returns the current setting of the internally generated payload fixed pattern.

Syntax SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe?

Response

<fixed pattern> (NR1-numeric)	description
Any integer in the range 0 to 255	The current setting of the payload (default = 00)

Dependencies None

Errors and Events None

Examples Query: SOURCE:DATA:TELECOM:PAYLOAD:PATTERN:UBYTE?

Response: 123

Related Commands SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe

SOURce:DATA:TELecom:OVERhead and POVerhead Subsystem

This section describes the commands and queries that set up the transport overhead and path overhead. Figure 2–12 shows the hierarchy tree for this subsystem.

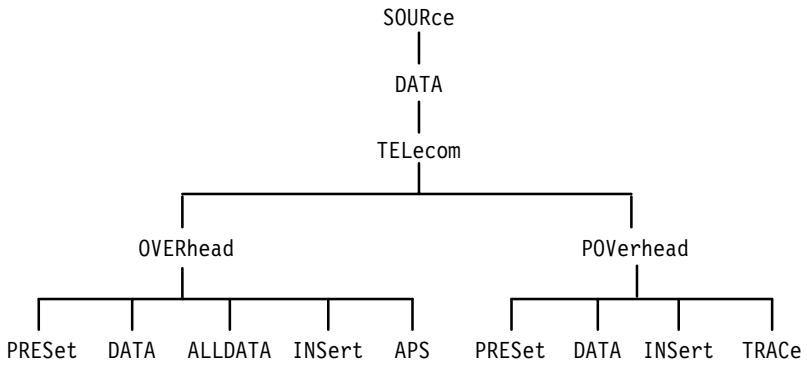


Figure 2–12: SOURce:DATA:TELecom:OVERhead and POVerhead Subsystem

Figures 2–14 and 2–15 list the bytes in the Transport and Path Overhead and the value of each byte after a *RST command is sent or a rate change occurs. As shown in Figure 2–13, each box can contain as many as three numbers: the overhead byte name in the upper left corner, the hexadecimal value of the byte at the bottom, and a circled number in the upper right corner. More information about these circled numbers is shown in the legend. General information is listed below the table in Notes.

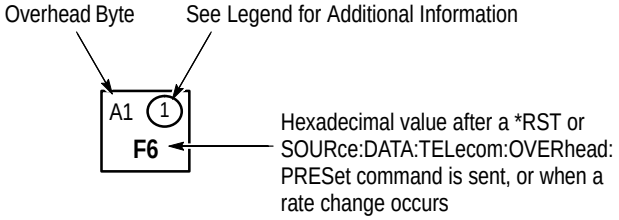


Figure 2–13: How to Read the Overhead Default Values Table

STS-1 Transport Overhead

A1 **F6**	A2 **28**	C1 (1) **01**
B1 (2) **HW**	E1 **00**	F1 **00**
D1 **00**	D2 **00**	D3 **00**
H1 (3) **HW**	H2 (3) **HW**	H3 **HW**
B2 **HW**	K1 **00**	K2 **00**
D4 **00**	D5 **00**	D6 **00**
D7 **00**	D8 **00**	D9 **00**
D10 **00**	D11 **00**	D12 **00**
Z1 **00**	Z2 **00**	E2 **00**

Path Overhead

J1 (4) **00**
B3 **HW**
C2 (5) **00**
G1 **00**
F2 **00**
H4 **HW**
Z3 **00**
Z4 **00**
Z5 **00**

Legend

(1) See Bellcore Specification TR-NWT-000253 for a description.

(2) Only the B1 byte in the first STS-1 channel will be set by the hardware; the rest will be set to 0.

(3) Default pointer value for H1 and H2 is hexadecimal 20A.
The s-bits of H1 are set to 00. The n-bits of H1 are set to 0110.
The default for H1 is 01100010.
The default for H2 is 00001010.

(4) The default for J1 is 64 nulls.

(5) The C2 value is set by mapping.

Notes

- All values are in hexadecimal.
- "HW" indicates that the hexadecimal value is determined dynamically by the hardware.
- For multiplexed signals, all bytes except B1, C1, and, possibly, Z2 are duplicated.
- Only one path overhead exists for all SONET rates.
- When a rate change occurs, the overhead will be reset to the above default values.

Figure 2–14: SONET STS-1 Overhead Default Values

STS-3c Transport Overhead

A1 **F6**	A1 **F6**	A1 **F6**	A2 **28**	A2 **28**	A2 **28**	C1 (1) **01**	NU **00**	NU **00**
B1 **HW**	– **00**	– **00**	E1 **00**	– **00**	– **00**	F1 **00**	NU **00**	NU **00**
D1 **00**	– **00**	– **00**	D2 **00**	– **00**	– **00**	D3 **00**	– **00**	– **00**
H1 (2) **HW**	H1 **93**	H1 **93**	H2 (2) **HW**	H2 **FF**	H2 **FF**	H3 **HW**	H3 **HW**	H3 **HW**
B2 **HW**	B2 **HW**	B2 **HW**	K1 **00**	– **00**	– **00**	K2 **00**	– **00**	– **00**
D4 **00**	– **00**	– **00**	D5 **00**	– **00**	– **00**	D6 **00**	– **00**	– **00**
D7 **00**	– **00**	– **00**	D8 **00**	– **00**	– **00**	D9 **00**	– **00**	– **00**
D10 **00**	– **00**	– **00**	D11 **00**	– **00**	– **00**	D12 **00**	– **00**	– **00**
Z1 **00**	Z1 **00**	Z1 **00**	Z2 **00**	Z2 **00**	Z2 (3) **00**	E2 **00**	NU **00**	NU **00**
Offset Value 0	1	2	0	1	2	0	1	2

Path Overhead

J1 (4) **00**
B3 **HW**
C2 (5) **00**
G1 **00**
F2 **00**
H4 **00**
Z3 **00**
Z4 **00**
Z5 **00**

Legend

(1) C1 in indicates the order of appearance of the within the frame. (SDH only)

(2) Default pointer value for H1 and H2 is hexadecimal 20A.
The s-bits of H1 are set to 10. The n-bits of H1 are set to 0110.
The default for H1 is 01101010.
The default for H2 is 00001010.

(3) The third M1 of an STS-3ccan be set by hardware Line FEBE (determined by error rate and type).

(4) The default for J1 is 64 nulls.

(5) The C2 value is set by mapping.

Notes

- All values are in hexadecimal.
- "NU" indicates a National Use Byte.
- "–" indicates an unnamed byte.
- "HW" indicates that the hexadecimal value is determined dynamically by the hardware.
- The offset value at the bottom of each column is used with the SOURce:DATA:TELecom:OVERhead:DATA and SENSe:DATA:TELecom:OVERhead:DATA commands (concatenated structures only).
- For multiplexed signals, all bytes except B1, C1, and possibly M1 are duplicated.
- Only one path overhead exists for all rates.
- When a rate change occurs, the overhead will be reset to the above default values.

Figure 2–15: SONET STS-3c Overhead Default Values

SOURce:DATA:TELecom:OVERhead:PRESet

This command resets the entire overhead to the default (see Figures 2–14 and 2–15 on pages 2–38 and 2–39 for the default values).

Syntax `SOURce:DATA:TELecom:OVERhead:PRESet`

Parameters None

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:OVERHEAD:PRESET`

Related Commands SOURce:DATA:TELecom:OVERhead:DATA

SOURce:DATA:TELecom:OVERhead:DATA

This command sets the bytes in the transport overhead. Bytes B1, B2, B3, H1, H2, and H3 are not accessible because they are controlled directly by the hardware.

Syntax `SOURce:DATA:TELecom:OVERhead:DATA <channel>,<byte>,<offset>, <value>`

Parameters

<channel> (NR1-numeric)	description
1	Rate is STS-1 or STS-3 rate with STS-3c structure
1 to 3	Rate is STS-3 with STS-1 structure
1 to 4	Rate is STS-12 with STS-3c structure
1 to 12	Rate is STS-12 with STS-1 structure
<byte> (discrete)	**description**
A1, A2, C1, E1, F1, D1, D2, D3, K1, K2, D4, D5, D6, D7, D8, D9, D10, D11, D12, Z1, Z2, E2	Only the bytes listed are available for selection

(continued on next page)

<offset> (NR1-numeric)	description
0	STS-1 structure
0 to 2	STS-3c structure
<value> (NR1-numeric)[1]	**description**
Any integer in the range 0 to 255 (hexadecimal 00 to FF)	The byte is set to this value

1 A hexadecimal value is also acceptable.

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:OVERHEAD:DATA 1,D3,0,#HAA`

Sets the D3 byte in the first overhead of an STS-12 signal to hexadecimal AA.

Related Commands None

SOURce:DATA:TELecom:OVERhead:DATA?

This query returns the value in the specified overhead byte from the transport overhead memory. You cannot set bytes B1, B2, B3, H1, H2, and H3 because they are controlled directly by the hardware.

Use <channel>, <byte>, and <offset> to address all named and unnamed bytes in the concatenated structures.

Syntax `SOURce:DATA:TELecom:OVERhead:DATA? <channel>,<byte>,<offset>`

Parameters

<channel> (NR1-numeric)	description
1	Rate is STS-1 or STS-3 rate with STS-3c structure
1 to 3	Rate is STS-3 with STS-1 structure
1 to 4	Rate is STS-12 with STS-3c structure
1 to 12	Rate is STS-12 with STS-1 structure

(continued on next page)

<byte> (discrete)	description
A1, A2, C1, E1, F1, D1, D2, D3, K1, K2, D4, D5, D6, D7, D8, D9, D10, D11, D12, Z1, Z2, E2	Only the bytes listed are available for selection

<offset> (NR1-numeric)	description
0	STS-1 structure
0 to 2	STS-3c structure

Response

<value> (NR1-numeric)	description
Any integer in the range 0 to 255	The byte is set to this value

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:OVERHEAD:DATA? 1,C1,0`

Response: `255`

Related Commands SOURce:DATA:TELecom:OVERhead:DATA

SOURce:DATA:TELecom:OVERhead:ALLData

This command allows 22 overhead bytes to be set at one time.

Syntax

```
SOURce:DATA:TELecom:OVERhead:ALLDATA <channel>,<offset>,<A1>,
<A2>,<C1>,<E1>,<F1>,<D1>,<D2>,<D3>,<K1>,<K2>,<D4>,<D5>,<D6>,<D7>,
<D8>,<D9>,<D10>,<D11>,<D12>,<Z1>,<Z2>,<E2>
```

Parameters

<channel> (NR1-numeric)	description
Any integer in the range 1 to 12	This value indicates the desired channel setting

<offset> (NR1-numeric)	description
Any integer in the range 0 to 2	This value indicates the desired offset

<A1>,<A2>,<C1>,<E1>,<F1>,<D1>, <D2>,<D3>,<K1>,<K2>,<D4>,<D5>,<D6>, <D7>,<D8>,<D9>,<D10>,<D11>,<D12>, <Z1>,<Z2>,<E2> (NR1-numeric)[1]	description
Any integer in the range 0 to 255 for each parameter (hexadecimal 00 to FF)	These values indicate the desired setting for each overhead byte

1 **A hexadecimal value is also acceptable.**

Dependencies <channel> and <offset> must be compatible with the current rate and structure settings.

Errors and Events 221, "Settings conflict; Parameter out of range"

Examples

```
SOURCE:DATA:TELECOM:OVER:ALLDATA 1,0,92,123,1,0,23,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0
```

This example sets the A1 byte to 92, the A2 byte to 123, the C1 byte to 1, the E1 byte to 23, and the rest of the bytes to 0 for channel 1.

Related Commands SOURce:DATA:TELecom:OVERhead:DATA

SOURce:DATA:TELecom:OVERhead:ALLData?

This query returns overhead data in a command form that can be used to set the available overhead bytes. One command with 24 parameters is produced. The first two parameters indicate channel and offset. The remaining 22 parameters are the data values for the overhead bytes in decimal number form.

Syntax

```
SOURce:DATA:TELecom:OVERhead:ALLData? <channel>,<offset>
```

Parameters

<channel> (NR1-numeric)	description
Any integer in the range 1 to 12	This value indicates the desired channel setting
<offset> (NR1-numeric)	**description**
Any integer in the range 0 to 2	This value indicates the desired offset

Response

<A1>,<A2>,<C1>,<E1>,<F1>,<D1>, <D2>,<D3>,<K1>,<K2>,<D4>,<D5>,<D6>, <D7>,<D8>,<D9>,<D10>,<D11>,<D12>, <Z1>,<Z2>,<E2> (NR1-numeric)	description
Any integer in the range 0 to 255 for each parameter	These values indicate the desired setting for each overhead byte

Dependencies None

Errors and Events 118, "Query not allowed"

Examples

Query: `SOURCE:DATA:TELECOM:OVERHEAD:ALLDATA? 1,0`

Response: `SOURCE:DATA:TELECOM:OVERHEAD:ALLDATA 1, 0, 92, 123, 1, 0, 23, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0`

Related Commands SOURce:DATA:TELecom:OVERhead:ALLDATA

SOURce:DATA:TELecom:OVERhead:INSert

This command controls the insertion of data into the overhead from an external protocol analyzer into the specific overhead bytes.

Syntax `SOURce:DATA:TELecom:OVERhead:INSert <insert>`

Parameters

<insert> (discrete)	description
NONE	Off (default)
SDCC	Section DCC
LDCC	Line DCC
F1	F1 byte

Dependencies You can insert data into the overhead or the path overhead by using the SOURce:DATA:TELecom:OVERhead:INSert and SOURce:DATA:TELecom:POVerhead:INSert commands. The last command sent applies.

Errors and Events None

Examples `SOURCE:DATA:TELECOM:OVERHEAD:INSERT SDCC`

Related Commands None

SOURce:DATA:TELecom:OVERhead:INSert?

This query returns the channel being inserted into the overhead from an external protocol analyzer.

Syntax `SOURce:DATA:TELecom:OVERhead:INSert?`

Response

<insert> (discrete)	description
NONE	Off (default)
SDCC	Section DCC
LDCC	Line DCC
F1	F1 byte

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:OVERHEAD:INSERT?`

Response: `F1`

Related Commands SOURce:DATA:TELecom:OVERhead:INSert

SOURce:DATA:TELecom:OVERhead:APS

This command simultaneously sets the K1 and K2 bytes in the same frame. Use the SENSe:DATA:TELecom:OVERhead:DATA? query to find out the value of the K1 and K2 bytes in the received signal.

Syntax `SOURce:DATA:TELecom:OVERhead:APS <APS value>`

Parameters

<APS value> (NR1-numeric)[1]	description
Any integer in the range 0 to 65535 (hexadecimal 0 to FFFF)	The 16-bit value of the K1 and K2 APS bytes

1 **A hexadecimal value is also acceptable.**

Dependencies None

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:OVERHEAD:APS #HFFFF
```

This example sets both K1 and K2 bytes to the maximum value (binary 1111111111111111).

```
SOURCE:DATA:TELECOM:OVERHEAD:APS #HFF00
```

This example sets the K1 byte to the maximum value (binary 11111111) and the K2 byte to 0 (binary 00000000).

Related Commands None

SOURce:DATA:TELecom:OVERhead:APS?

This query returns the 16-bit value of the K1 and K2 bytes.

Syntax

```
SOURce:DATA:TELecom:OVERhead:APS?
```

Response

<APS value> (NR1-numeric)	description
Any integer in the range 0 to 65535	The 16-bit value of the K1 and K2 APS bytes

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:OVERHEAD:APS?`

Response: `65535`

Related Commands None

SOURce:DATA:TELecom:POVerhead:PRESet

This command resets the path overhead to the default (see Tables 2–14 and 2–15 on pages 2–38 and 2–39Figure 2–15 on page 2–39 for the default values).

Syntax `SOURce:DATA:TELecom:POVerhead:PRESet`

Parameters None

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:POVERHEAD:PRESET`

Related Commands SOURce:DATA:TELecom:POVerhead:DATA

SOURce:DATA:TELecom:POVerhead:DATA

This command sets the bytes in the path overhead.

Syntax `SOURce:DATA:TELecom:POVerhead:DATA <byte>,<value>`

Parameters

<byte> (discrete)	description
C2, F2, Z3, Z4, Z5	Only the bytes listed are available for selection[1]

<value> (NR1-numeric)[2]	description
Any integer in the range 0 to 255 (hexadecimal 00 to FF)	The selected byte is set to this value

1 **The J1 path trace overhead byte is controlled through the SOURce:DATA:TELecom:POVerhead:TRACe command.**

2 **A hexadecimal value is also acceptable.**

Dependencies This command is ignored if SOURce:DATA:TELecom:POVerhead:INSert is set to F2.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:POVERHEAD:DATA D1,#H55
```

Related Commands SOURce:DATA:TELecom:POVerhead:TRACe
SOURce:DATA:TELecom:PAYLoad:MAPPing (sets the C2 byte)

SOURce:DATA:TELecom:POVerhead:DATA?

This query returns the value in the specified path overhead bytes.

Syntax

```
SOURce:DATA:TELecom:POVerhead:DATA? <byte>
```

Parameters

<byte> (discrete)	description
C2, F2, G1, Z3, Z4, Z5	Only the bytes listed are available for selection

Response

<value> (NR1-numeric)	description
Any integer in the range 0 to 255	The selected byte is set to this value

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POVERHEAD:DATA? C2`

Response: `255`

Related Commands SOURce:DATA:TELecom:POVerhead:DATA

SOURce:DATA:TELecom:POVerhead:INSert

This command controls the insertion of data into the path overhead from an external protocol analyzer into the specific overhead bytes.

Syntax `SOURce:DATA:TELecom:POVerhead:INSert <path insert>`

Parameters

<path insert> (discrete)	description
NONE	Off
F2	F2 byte

Dependencies You can insert data into the overhead or the path overhead by using the SOURce:DATA:TELecom:OVERhead:INSert and SOURce:DATA:TELecom:POVerhead:INSert commands. The last command sent applies.

Errors and Events None

Examples `SOURCE:DATA:TELECOM:POVERHEAD:INSERT F2`

Related Commands None

SOURce:DATA:TELecom:POVerhead:INSert?

This query returns the channel being inserted into the path overhead from an external protocol analyzer.

Syntax `SOURce:DATA:TELecom:POVerhead:INSert?`

Response

<path insert> (discrete)	description
NONE	Off
F2	F2 byte

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POVERHEAD:INSERT?`

Response: `F2`

Related Commands SOURce:DATA:TELecom:POVerhead:INSert

SOURce:DATA:TELecom:POVerhead:TRACe

This command sets the path trace overhead bytes that appear in J1 as a repeating 64-byte sequence. The string must not exceed 64 ASCII characters in length. Unprintable characters will be accepted and inserted directly.

Syntax `SOURce:DATA:TELecom:POVerhead:TRACe <path trace>`

Parameters

<path trace> (string)	description
Length is a maximum of 64 bytes; if length is less than 64 bytes, the buffer is padded with nulls to a length of 64 bytes; the string will be terminated with a CR/LF (carriage return/line feed)	The J1 byte is set to this value (default is 64 null characters)

Dependencies None

Errors and Events 223, “Too much data; Path trace string truncated”

Examples `SOURCE:DATA:TELECOM:POVERHEAD:TRACE "TESTING 1 . 2 . 3"`

Related Commands SOURce:DATA:TELecom:POVerhead:DATA?

SOURce:DATA:TELecom:POVerhead:TRACe?

This query returns the current path trace string that repeats in the J1 byte as a 64-byte repeating sequence.

Syntax `SOURce:DATA:TELecom:POVerhead:TRACe?`

Response

<path trace> (string)	description
Length is a maximum of 64 bytes; if length is less than 64 bytes, the buffer is padded with nulls to a length of 64 bytes	The J1 byte is set to this value (default is 64 null characters)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POVERHEAD:TRACE?`

Response: `"THIS IS A TEST"`

Related Commands SOURce:DATA:TELecom:POVerhead:TRACe

SOURce:DATA:TELecom:ERRor, ALARm, and FAILure Subsystem

This section describes the commands and queries that control abnormal conditions such as errors, alarms, and failures in the transmitted signal. Figure 2–16 shows the hierarchy tree for this subsystem.

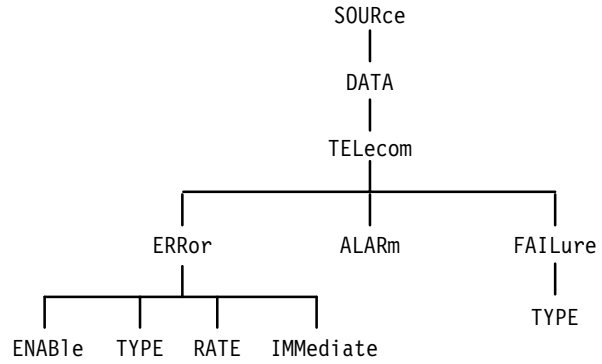


Figure 2–16: SOURce:DATA:TELecom:ERRor, ALARm, and FAILure Subsystem

SOURce:DATA:TELecom:ERRor:ENABle

This command enables errors to be inserted into the output signal.

Syntax `SOURce:DATA:TELecom:ERRor:ENABle <error rate state>`

Parameters

<error rate state> (boolean)	description
0 or OFF	Error rate disabled (default)
1 or ON	Errors specified by rate

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:ERROR:ENABLE 0`

Related Commands SOURce:DATA:TELecom:ERRor:RATE

SOURce:DATA:TELecom:ERRor:ENABle?

This query returns the current enable setting of the error rate.

Syntax `SOURce:DATA:TELecom:ERRor:ENABle?`

Response

<error rate state> (boolean)	description
0	Error rate off (OFF)
1	Errors specified by rate (ON)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:ERROR:ENABLE?`

Response: 0

Related Commands SOURce:DATA:TELecom:ERRor:MODE

SOURce:DATA:TELecom:ERRor:TYPE

This command selects the error type.

Syntax `SOURce:DATA:TELecom:ERRor:TYPE <error type>`

Parameters

<error type> (discrete)	description
SCV	Section B1 BIP error; B1 will be errored across all bits (default)
LCV	Line B2 BIP error; B2 will be errored across all bits
PCV	Path B3 BIP error; the active channel B3 will be errored across all bits
PFEBe	Path Far End Block Error (path FEBE at specified rate); a value of 1 is inserted in the G1 byte when the SOURce:DATA:TELecom:ERRor:IMMediate command is given

(continued on next page)

<error type> (discrete)	description
DATA	Payload data bit error (payload data will be errored but B3 will not)
TRIButary	Allows selection of tributary errors (Add/ Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:ERROR:TYPE SCV
```

Related Commands SOURce:DATA:TELecom:ERRor:RATE
SOURce:DATA:TELecom:TRIButary:ERRor

SOURce:DATA:TELecom:ERRor:TYPE?

This query returns the current setting of the selected error type.

Syntax

```
SOURce:DATA:TELecom:ERRor:TYPE?
```

Response

<error type> (discrete)	description
SCV	Section B1 BIP error; B1 will be errored across all bits (default)
LCV	Line B2 BIP error; B2 will be errored across all bits
PCV	Path B3 BIP error; the active channel B3 will be errored across all bits
PFEBe	Path Far End Block Error (path FEBE at specified rate); a value of 1 appears in the G1 byte if the SOURce:DATA:TELecom: ERRor:IMMediate command has been given
DATA	Payload data bit error (payload data will be errored but B3 will not)
TRIButary	Tributary errors selected (Add/Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:ERROR:TYPE?`

Response: `SCV`

Related Commands SOURce:DATA:TELecom:ERRor:TYPE

SOURce:DATA:TELecom:ERRor:RATE

This command selects the error rate. Resolution is limited to one digit. For example, 1E-6, 2E-9, and 1E-3 are valid values; 1.43E-4 and 2.7E-9 are not valid values. Invalid error rates will be changed to the nearest valid value. For example, 1.25E-5 (too many digits) will be changed to 1E-5, 1E-20 (below minimum) will be changed to 1E-10 (minimum), and 1 (above maximum) will be changed to 1E-3 (maximum).

To disable error generation at any specified rate, use the SOURce:DATA: TELecom:ERRor:ENABle OFF command.

Syntax `SOURce:DATA:TELecom:ERRor:RATE <error rate>`
(see Tables 2–5 and 2–6 for <error rate> limits)

Table 2–5: Error Insertion Rate Limits for SOURce:DATA:TELecom:ERRor:RATE

If rate and structure set to:	If error type set to SCV	If error type set to LCV	If error type set to PCV	If error type set to PFEBE	If error type set to DATA
STS1 rate and STS1 structure	1E–10 to 1E–3	1E–10 to 1E–3	1E–10 to 1E–3	1E–10 to 1E–3	1E–10 to 1E–3
STS3 rate and STS1 structure	1E–10 to 1E–4	1E–10 to 1E–4	1E–10 to 1E–3	1E–10 to 1E–3	1E–10 to 1E–3
STS3 rate and STS3c structure	1E–10 to 1E–4	1E–10 to 1E–4	1E–10 to 1E–4	1E–10 to 1E–4	1E–10 to 1E–3
STS12 rate and STS1 structure	1E–10 to 1E–5	1E–10 to 1E–4	1E–10 to 1E–3	1E–10 to 1E–3	1E–10 to 1E–3
STS12 rate and STS3c structure	1E–10 to 1E–5	1E–10 to 1E–4	1E–10 to 1E–4	1E–10 to 1E–4	1E–10 to 1E–3

The table lists the minimum and maximum rates.

All error rates are NR3-numeric.

Table 2–6: Error Insertion Rate Limits for SOURce:DATA:TELecom:ERRor:RATE (Add/Drop/Test Option Only)

If rate set to:	If error type set to VTBIP	If error type set to VTFEBE	If error type set to PARITY	If error type set to CRC	If error type set to FRAME	If error type set to DATA
DS1	1E–10 to 1E–3	1E–10 to 1E–3	not applicable	1E–8 to 1E–4	1E–5 to 1E–2	1E–8 to 1E–2
DS3	not applicable	not applicable	1E–9 to 1E–4	not applicable	1E–7 to 1E–2	1E–9 to 1E–2

The table lists the minimum and maximum rates.

All error rates are NR3-numeric.

Dependencies None

Errors and Events 500, “Execution warning; Numeric value greater than maximum limit”

500, “Execution warning; Numeric value less than minimum limit”

Examples

```
SOURCE:DATA:TELECOM:ERROR:RATE 1E-6
```

Related Commands SOURce:DATA:TELecom:ERRor:TYPE
SOURce:DATA:TELecom:TRIButary:ERRor

SOURce:DATA:TELecom:ERRor:RATE?

This query returns the current setting of the error rate.

Syntax

```
SOURce:DATA:TELecom:ERRor:RATE?
```

Response

<error rate> (NR3-numeric)	description
See tables 2–5 and 2–6 on page 2–55 for valid ranges	Error rate is set to a value in this range

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:ERROR:RATE?`

Response: `1E-10`

Related Commands

SOURce:DATA:TELecom:ERRor:RATE

SOURce:DATA:TELecom:ERRor:IMMediate

This command is used to force an error insertion. The error is defined by SOURce:DATA:TELecom:DATA:ERRor:TYPE.

Syntax

`SOURce:DATA:TELecom:ERRor:IMMediate`

Parameters

None

Dependencies

SOURce:DATA:TELecom:ERRor:ENABle must be set to ON.

Errors and Events

None

Examples

`SOURCE:DATA:TELECOM:ERROR:IMMEDIATE`

Related Commands

SOURce:DATA:TELecom:ERRor:TYPE
SOURce:DATA:TELecom:TRIButary:ERRor

SOURce:DATA:TELecom:ALARm

This command selects an alarm to transmit.

Syntax

`SOURce:DATA:TELecom:ALARm <alarm>`

Parameters

<alarm> (discrete)	description
NONE	No alarms (default)
LAIS	Line AIS
PAIS	Path AIS
LFERf	Line FERF

(continued on next page)

<alarm> (discrete)	description
PFERf	Path FERF
TRIButary	Allows selection of tributary alarms (Add/ Drop/Test Option Only)

Dependencies SOURce:DATA:TELecom:FAILure:TYPE must be set to NONE for this command to apply.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:ALARM LAIS
```

Related Commands SOURce:DATA:TELecom:TRIButary:ALARm

SOURce:DATA:TELecom:ALARm?

This query returns the current setting of the transmitted alarm type.

Syntax

```
SOURce:DATA:TELecom:ALARm?
```

Response

<alarm> (discrete)	description
NONE	No alarms (default)
LAIS	Line AIS
PAIS	Path AIS
LFERf	Line FERF
PFERf	Path FERF
TRIButary	Tributary alarms selected (Add/Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:ALARM?`

Response: `LAIS`

Related Commands SOURce:DATA:TELecom:ALARm

SOURce:DATA:TELecom:FAILure:TYPE

This command selects a failure type to transmit. Selecting a failure type overrides all errors and alarms.

Syntax `SOURce:DATA:TELecom:FAILure:TYPE <failure>`

Parameters

<failure> (discrete)	description
NONE	No failures (default)
LOSignal	Loss of Signal (disconnects the output signal)
LOFrame	Loss of Frame (changes the most significant bit of A1 resulting in a hexadecimal value of 76)
LOPointer	Loss of Pointer (generates continuous NDFs)
TRIButary	Allows selection of tributary failures (Add/Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:FAILURE:TYPE LOS`

Related Commands SOURce:DATA:TELecom:TRIButary:FAILure

SOURce:DATA:TELecom:FAILure:TYPE?

This query returns the current setting of the failure type.

Syntax `SOURce:DATA:TELecom:FAILure:TYPE?`

Response

<failure> (discrete)	description
NONE	No failures (default)
LOSignal	Loss of Signal (output signal is disconnected)
LOFrame	Loss of Frame (A1 is set to hexadecimal 76)
LOPointer	Loss of Pointer (continuous NDFs generated)
TRIButary	Tributary failures selected (Add/Drop/Test Option Only)

Dependencies

None

Errors and Events

None

Examples

Query: `SOURCE:DATA:TELECOM:FAILURE:TYPE?`

Response: `LOSIGNAL`

Related Commands

SOURce:DATA:TELecom:FAILure:TYPE

SOURce:DATA:TELecom:POINter Subsystem

This section describes the commands and queries that adjust pointers. Figure 2–17 shows the hierarchy tree for this subsystem.

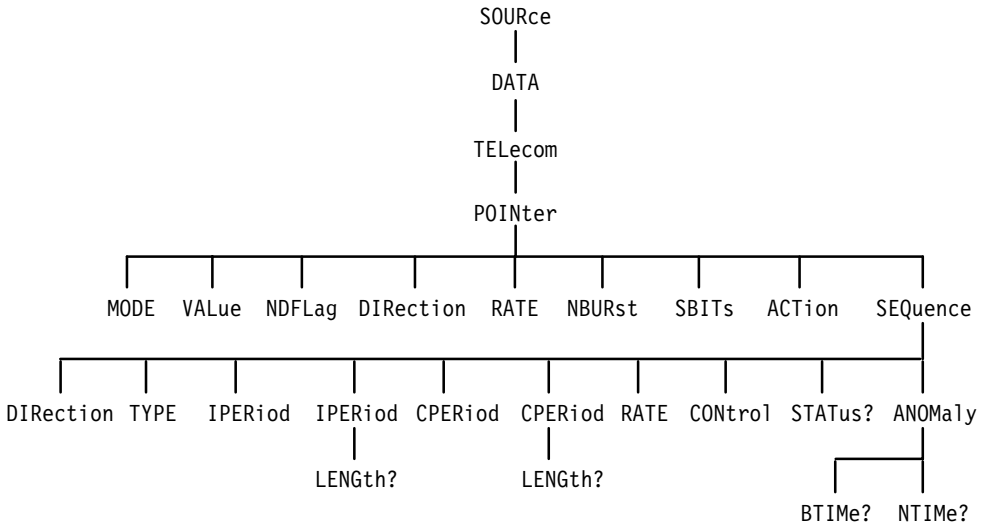


Figure 2–17: SOURce:DATA:TELecom:POINter Subsystem

SOURce:DATA:TELecom:POINter:MODE

This command controls the pointer manipulation modes.

Syntax `SOURce:DATA:TELecom:POINter:MODE <mode>`

Parameters

<mode> (discrete)	description
MANual	Pointers are controlled by SOURce:DATA:TELecom:POINter:VALue and SOURce:DATA:TELecom:NDFlag (default)
SINGle	When the SOURce:DATA:TELecom:POINter:ACTion command is given, pointer adjustments will alternately increment and decrement
BURSt	When the SOURce:DATA:TELecom:POINter:ACTion command is given, a burst of pointer adjustments is sent at the maximum rate (1 in 4 frames) and with a count defined by SOURce:DATA:TELecom:POINter:NBURst

(continued on next page)

<mode> (discrete)	description
FOFFset	Frequency offset pointers are controlled by the SOURce:CLOCk:OFFSet: commands
CONTinuous	Pointers are continuously adjusted according to the SOURce:DATA:TELecom: POINter:RATE and SOURce:DATA: TELecom:POINter:DIRection commands
TRIButary	Pointers are controlled by the SOURce:DATA: TELecom:TRIButary:POINter subsystem (Add/Drop/Test Option Only)
SEQuence	Pointers are stressed according to sequences defined in T1.105.03–1994 or G.783.

Dependencies None

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:POINTER:MODE FOFFSET
```

Related Commands None

SOURce:DATA:TELecom:POINter:MODE?

This query returns the current setting of the pointer mode.

Syntax

```
SOURce:DATA:TELecom:POINter:MODE?
```

Response

<mode> (discrete)	description
MANual	Pointers are controlled by SOURce:DATA: TELecom:POINter:VALue and SOURce:DATA: TELecom:NDFlag (default)
SINGle	When the SOURce:DATA:TELecom: POINter:ACTion command is given, pointer adjustments will alternately increment and decrement

(continued on next page)

<mode> (discrete)	description
BURSt	When the SOURce:DATA:TELecom:POINter:ACTion command is given, a burst of pointer adjustments is sent at the maximum rate (1 in 4 frames) and with a count defined by SOURce:DATA:TELecom:POINter:NBURst
FOFFset	Frequency offset pointers are controlled by the SOURce:CLOCk:OFFSet: commands
CONTinuous	Pointers are continuously adjusted according to the SOURce:DATA:TELecom:POINter:RATE and SOURce:DATA:TELecom:POINter:DIRection commands
TRIButary	Pointers are controlled by the SOURce:DATA:TELecom:TRIButary:POINter subsystem (Add/Drop/Test Option Only)
SEQuence	Pointers are stressed according to sequences defined in T1.105.03–1994 or G.783.

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:MODE?`

Response: `MANUAL`

Related Commands SOURce:DATA:TELecom:POINter:MODE

SOURce:DATA:TELecom:POINter:VALue

This command directly sets the pointer value. To obtain an illegal pointer value, use a value greater than 782. If SOURce:DATA:TELecom:POINter:NDFlag is set to ON, a New Data Flag (NDF) is sent with each new value received.

Syntax `SOURce:DATA:TELecom:POINter:VALue <pointer value>`

Parameters

<pointer value> (NR1-numeric)	description
Any integer in the range 0 to 1023	Pointer set to this value (default = 522)

Dependencies SOURce:DATA:TELecom:POINter:MODE must be set to MANual for this command to apply.

Errors and Events None

Examples `SOURCE:DATA:TELECOM:POINTER:VALUE 10`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:NDFlag

SOURce:DATA:TELecom:POINter:VALue?

This query returns the current pointer value being transmitted.

Syntax `SOURce:DATA:TELecom:POINter:VALue?`

Response

<pointer value> (NR1-numeric)	description
Any integer in the range 0 to 1023	Pointer set to this value (default = 522)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:VALUE?`

Response: `310`

Related Commands SOURce:DATA:TELecom:POINter:VALue

SOURce:DATA:TELecom:POINter:NDFLag

This command controls the generation of a New Data Flag (NDF) when pointer adjustments occur.

Syntax `SOURce:DATA:TELecom:POINter:NDFLag <NDF state>`

Parameters

<NDF state> (boolean)	description
1 or ON	On (default)
0 or OFF	Off

Dependencies SOURce:DATA:TELecom:POINter:MODE must be set to MANual for this command to apply.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:POINTER:NDFLag ON
```

Related Commands SOURce:DATA:TELecom:POINter:VALue

SOURce:DATA:TELecom:POINter:NDFLag?

This query returns the current setting of the New Data Flag (NDF) generator.

Syntax

```
SOURce:DATA:TELecom:POINter:NDFLag?
```

Response

<NDF state> (boolean)	description
1	On (default)
0	Off

Dependencies None

Errors and Events None

Examples Query:

```
SOURCE:DATA:TELECOM:POINTER:NDFLAG?
```

Response: 0

Related Commands SOURce:DATA:TELecom:POINter:NDFLag

SOURce:DATA:TELecom:POINter:DIRection

This command sets the direction of continuous pointer adjustments.

Syntax

```
SOURce:DATA:TELecom:POINter:DIRection <direction>
```

Parameters

<direction> (discrete)	description
ALTernate	Pointer adjustments alternate between up and down (default)
DOWN	Pointers adjusted down
UP	Pointers adjusted up

Dependencies SOURce:DATA:TELecom:POINTer:MODE must be set to CONTinuous for this command to apply.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:POINTER:DIRECTION UP
```

Related Commands SOURce:DATA:TELecom:POINter:RATE

SOURce:DATA:TELecom:POINter:DIRection?

This query returns the current setting for continuous pointer adjustment direction.

Syntax

```
SOURce:DATA:TELecom:POINter:DIRection?
```

Response

<direction> (discrete)	description
ALTernate	Pointer adjustments alternate between up and down (default)
DOWN	Pointers adjusted down
UP	Pointers adjusted up

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:DIRECTION?`

Response: `UP`

Related Commands SOURce:DATA:TELecom:POINter:DIRection

SOURce:DATA:TELecom:POINter:RATE

This command sets the continuous pointer adjustment rate.

Syntax `SOURce:DATA:TELecom:POINter:RATE <rate>`

Parameters

<rate> (NR1-numeric)	description
Any integer in the range 2 to 10,000 ms (resolution of 1 ms)	The pointer adjustment rate is set to this value

Dependencies SOURce:DATA:TELecom:POINTer:MODE must be set to CONTinuous for this command to apply.

Errors and Events None

Examples `SOURCE:DATA:TELECOM:POINTER:RATE 10`

Related Commands SOURce:DATA:TELecom:POINter:DIRection
SOURce:DATA:TELecom:POINter:MODE

SOURce:DATA:TELecom:POINter:RATE?

This query returns the current setting of the continuous pointer adjustment rate.

Syntax `SOURce:DATA:TELecom:POINter:RATE?`

Response

<rate> (NR1-numeric)	description
Any integer in the range 2 to 10,000 ms (resolution of 1 ms)	The pointer adjustment rate is set to this value

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:RATE?`

Response: 3

Related Commands SOURce:DATA:TELecom:POINter:RATE

SOURce:DATA:TELecom:POINter:NBURst

This command sets the number of pointer adjustments in a burst of pointer adjustments. The SOURce:DATA:TELecom:POINTer:ACTion command controls when the burst occurs.

Syntax `SOURce:DATA:TELecom:POINter:NBURst <pointer burst number>`

Parameters

<pointer burst number> (NR1-numeric)	description
Any integer in the range 2 to 8	This value determines the number of pointer adjustments in a burst of pointer adjustments (default = 2)

Dependencies SOURce:DATA:TELecom:POINter:MODE must be set for BURSt for this command to apply.

Errors and Events None

Examples `SOURCE:DATA:TELECOM:POINTER:NBURST 2`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:ACTion

SOURce:DATA:TELecom:POINter:NBURst?

This query returns the number of pointer adjustments in a burst of pointer adjustments.

Syntax `SOURce:DATA:TELecom:POINter:NBURst?`

Response

<pointer burst number> (NR1-numeric)	description
Any integer in the range 2 to 8	This value determines the number of pointer adjustments in a burst of pointer adjustments (default = 2)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:NBURST?`

Response: 3

Related Commands SOURce:DATA:TELecom:POINter:NBURst

SOURce:DATA:TELecom:POINter:SBITs

This command sets static value of the S-bits (bits 5 and 6) in the H1 byte.

Syntax `SOURce:DATA:TELecom:POINter:SBITs <pointer sbits>`

Parameters

<pointer sbits> (NR1-numeric)	description
Any integer in the range 0 to 3	This value is the S-bit in the H1 byte (default = 0)

Dependencies None

Errors and Events 500, “Execution warning; Numeric value greater than maximum limit”

Examples

```
SOURCE:DATA:TELECOM:POINTER:SBITS 0
```

Related Commands None

SOURce:DATA:TELecom:POINter:SBITs?

This query returns the static value of the S-bits (bits 5 and 6) in the H1 byte.

Syntax

```
SOURce:DATA:TELecom:POINter:SBITs?
```

Response

<pointer sbits> (NR1-numeric)	description
Any integer in the range 0 to 3	This value is the S-bit in the H1 byte (default = 0)

Dependencies None

Errors and Events None

Examples

Query:

```
SOURCE:DATA:TELECOM:POINTER:SBITS?
```

Response: 3

Related Commands SOURce:DATA:TELecom:POINter:SBITs

SOURce:DATA:TELecom:POINter:ACTion

This command invokes a pointer adjustment for SONET or tributary signals.

Syntax

```
SOURce:DATA:TELecom:POINter:ACTion
```

Parameters None

Dependencies SOURce:DATA:TELecom:POINter:MODE or if the SOURce:DATA:TELecom:POINter:MODE is TRIButary, SOURce:DATA:TELecom:

TRIButary:POINter:MODE must be set to SINGle or BURst for this command to apply.

Errors and Events

221, “Settings conflict; Mode must be single or burst”
200, “Execution error; Pointer burst active, request ignored”

Examples

```
SOURCE:DATA:TELECOM:POINTER:ACTION
```

Related Commands

SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:NBURst
SOURce:DATA:TELecom:TRIButary:POINter:MODE
SOURce:DATA:TELecom:TRIButary:POINter:NBURst

SOURce:DATA:TELecom:POINter:SEQuence:DIRection

This command sets the pointer sequence movement direction.

Setting this parameter when a sequence is running returns an error.

Syntax

```
SOURce:DATA:TELecom:POINter:SEQuence:DIRection
<pointer seq direction>
```

Parameters

<pointer seq direction> (discrete)	description
DOWN	(default)
UP	

Dependencies

Pointer sequences must not be running.

Errors and Events

221, “Settings conflict; stop sequences before setting the direction”

Examples

```
SOURCE:DATA:TELECOM:POINTER:SEQUENCE:DIRECTION UP
SOUR:DATA:TEL:POIN:SEQ:DIR DOWN
```

Related Commands

SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:POINter:SEQuence:DIRection?

This query returns the direction for pointer sequence movements.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:DIRection?`

Parameters

<pointer seq direction> (discrete)	description
DOWN	(default)
UP	

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:DIRECTION?`

Response: `UP`

Related Commands SOURce:DATA:TELecom:POINter:SEQuence:DIRection

SOURce:DATA:TELecom:POINter:SEQuence:RATE

This command sets the pointer sequence movement rate in milliseconds.

Setting this parameter when a sequence is running returns an error.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:RATE <pointer seq rate>`

Parameters

<pointer seq rate> (NR1-numeric)	description
Any integer in the range 34 to 30,000 ms	Resolution is 1 ms (default = 34 ms)

Dependencies Pointer sequences must not be running. Some sequence types (such as single, burst, phase, sinalt, and dblalt) set the rate to 30,000 ms. With these types, you cannot change the rate.

Errors and Events 221, “Settings conflict; commanded to start but not in sequence mode”

Examples

```
SOURCE:DATA:TELECOM:POINTER:SEQUENCE:RATE 50
SOUR:DATA:TEL:POIN:SEQ:RATE 1000
```

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
SOURce:DATA:TELecom:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:POINter:SEQuence:RATE?

This query returns the rate for pointer sequence movements in milliseconds.

Syntax

```
SOURce:DATA:TELecom:POINter:SEQuence:RATE?
```

Parameters

<pointer seq rate> (discrete)	description
Any integer in the range 34 to 30,000 ms	Resolution is 1 ms (default = 34 ms)

Dependencies Some sequence types (like single, burst, phase, sinalt, and dblalt) set the rate to 30,000 ms. With these types, you cannot change the rate.

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:RATE?`

Response: `34`

Related Commands SOURce:DATA:TELecom:POINter:SEQuence:RATE

SOURce:DATA:TELecom:POINter:SEQuence:TYPE

This command sets the pointer sequence type.

Setting this parameter when a sequence is running returns an error.

Syntax

```
SOURce:DATA:TELecom:POINter:SEQuence:TYPE <pointer seq type>
```

Parameters

<pointer seq type> (discrete)	description	standard
SINGle	Single pointer adjustment	ANSI
BURSt	Burst pointer adjustment	ANSI
PHASE	Phase transient pointer adjustment	ANSI
P873	Periodic 87–3 pointer adjustment	ANSI
P873CAN	Periodic 87–3 with cancel	ANSI
P873ADD	Periodic 87–3 with add	ANSI
PCONtinuous	Periodic continuous pointer adjustment	ANSI
PCONCAN	Periodic continuous with cancel	ANSI
PCONADD	Periodic continuous with add	ANSI
SINALT	Single alternating pointer	ITU-T
DBLALT	Double alternating pointer	ITU-T

Dependencies Pointer sequences must not be running.

Errors and Events 221, “Settings conflict; stop sequences before setting the type”

Examples

```
SOURCE:DATA:TELECOM:POINTER:SEQUENCE:TYPE BURST
SOUR:DATA:TEL:POIN:SEQ:TYPE BURS
```

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:POINter:SEQuence:TYPE?

This query returns the pointer sequence type.

Syntax

```
SOURce:DATA:TELecom:POINter:SEQuence:TYPE?
```

Response

<pointer seq type> (discrete)	description
SINGle	Single pointer adjustment
BURSt	Burst pointer adjustment
PHASE	Phase transient pointer adjustment

(continued on next page)

<pointer seq type> (discrete)	description
P873	Periodic 87–3 pointer adjustment
P873CAN	Periodic 87–3 with cancel
P873ADD	Periodic 87–3 with add
PCONtinuous	Periodic continuous pointer adjustment
PCONCAN	Periodic continuous with cancel
PCONADD	Periodic continuous with add
SINALT	Single alternating pointer
DBLALT	Double alternating pointer

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:TYPE?`

Response: `PCONCAN`

Related Commands SOURce:DATA:TELecom:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:POINter:SEQuence:IPERiod

This command enables or disables the pointer sequence initialization period.

Setting this parameter when a sequence is running returns an error.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:IPERiod <pointer seq init>`

Parameters

<pointer seq init> (discrete)	description
0 (or OFF)	Disables the pointer sequence initialization period
1 (or ON)	Enables the pointer sequence initialization period (default)

Dependencies Pointer sequences must not be running.

Errors and Events 221, “Settings conflict; commanded to start but not in sequence mode”

Examples
```
SOURCE:DATA:TELECOM:POINTER:SEQUENCE:IPERIOD ON
```

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:POINter:SEQuence:IPERiod?

This query returns the pointer sequence initialization period enable/disable condition.

Syntax
```
SOURce:DATA:TELecom:POINter:SEQuence:IPERiod?
```

Response

<pointer seq init> (discrete)	description
0	Disabled
1	Enabled (default)

Dependencies None

Errors and Events None

Examples Query:
```
SOURCE:DATA:TELECOM:POINTER:SEQUENCE:IPERIOD?
```
Response:
```
0
```

Related Commands SOURce:DATA:TELecom:POINter:SEQuence:IPERiod

SOURce:DATA:TELecom:POINter:SEQuence:IPERiod:LENGth?

This query returns the pointer sequence initialization period in seconds.

Syntax
```
SOURce:DATA:TELecom:POINter:SEQuence:IPERiod:LENGth?
```

Response

<pointer seq init> (NR1-numeric)	description
Any integer	Pointer sequence initialization period in seconds

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:IPERIOD:LENGTH`

Response: 30

Related Commands SOURce:DATA:TELecom:POINter:SEQuence:IPERiod
SOURce:DATA:TELecom:POINter:SEQuence:RATE
SOURce:DATA:TELecom:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:POINter:SEQuence:CPERiod

This command enables or disables the pointer sequence cool down period.

Setting this parameter when a sequence is running returns an error.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:CPERiod <pointer seq init>`

Parameters

<pointer seq init> (discrete)	description
0 (or OFF)	Disables the pointer sequence cool down period
1 (or ON)	Enables the pointer sequence cool down period (default)

Dependencies Pointer sequences must not be running.

Errors and Events 221, “Settings conflict; commanded to start but not in sequence mode”

Examples `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:CPERIOD OFF`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:POINter:SEQuence:CPERiod?

This query returns the pointer sequence cool down period enable/disable condition.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:CPERiod?`

Response

<pointer seq init> (discrete)	description
0	Disabled
1	Enabled (default)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:CPERIOD?`

Response: 0

Related Commands SOURce:DATA:TELecom:POINter:SEQuence:CPERiod

SOURce:DATA:TELecom:POINter:SEQuence:CPERiod:LENGth?

This query returns the pointer sequence cool down period in seconds.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:CPERiod:LENGth?`

Response

<pointer seq init> (NR1-numeric)	description
Any integer	Pointer sequence cool down period in seconds

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:CPERIOD:LENGTH`

Response: 675

Related Commands SOURce:DATA:TELecom:POINter:SEQuence:CPERiod
SOURce:DATA:TELecom:POINter:SEQuence:RATE
SOURce:DATA:TELecom:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

This command controls pointer sequences and tributary pointer sequences.

A sequence begins with an initialization or cool down period, if enabled, and then enters operation. You may stop a pointer sequence at any time.

Syntax

```
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
<pointer seq control>
```

Parameters

<pointer seq control> (discrete)	description
STARt	Starts a pointer sequence
STOP	Stops a pointer sequence

Dependencies POINTer:MODE SEQuence or POINter:MODE TRIButary and TRIButary:POINter:MODE SEQuence must be selected.

Errors and Events 221, “Settings conflict; commanded to start but not in sequence mode”

Examples `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:CONtrol START`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:STATUS?

SOURce:DATA:TELecom:POINter:SEQuence:CONTrol?

This query returns the pointer sequences control status.

A sequence begins with an initialization and cool down period, and then enters operation. You may stop a pointer sequence at any time.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:CONTrol?`

Response

<pointer seq control> (discrete)	description
STARt	Starts a pointer sequence
STOP	Stops a pointer sequence

Dependencies POINTer:MODE SEQuence or POINter:MODE TRIButary and TRIButary:POINter:MODE SEQuence must be selected.

Errors and Events 221, “Settings conflict; commanded to start but not in sequence mode”

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:CONtrol?`

Response: `START`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
SOURce:DATA:TELecom:POINter:SEQuence:STATUS?

SOURce:DATA:TELecom:POINter:SEQuence:STATus?

This query returns the pointer sequence status.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:STATus?`

Response

<pointer seq status> (discrete)	description
STOPPED	Not running sequences
INITIALizing	Initialization period

(continued on next page)

<pointer seq status> (discrete)	description
COOLdown	Cool down period
OPERating	Running sequences

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:STATUS?`

Response: `INIT`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol?

SOURce:DATA:TELecom:POINter:SEQuence:ANOMaly:BTIMe?

This query returns the pointer sequence time between anomalies in seconds.

Sequences do not have to be running. This calculation is based upon sequence type and rate.

Syntax `SOURce:DATA:TELecom:POINter:SEQuence:ANOMaly:BTIMe?`

Response

<pointer seq btime> (NR1-numeric)	description
Any positive integer	Time between anomalies in seconds
–1	Returned for sequences with an invalid BTIME. Example: continuous without anomalies

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:ANOMALY:BTIME?`

Response: `30`

Related Commands

SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
SOURce:DATA:TELecom:POINter:SEQuence:TYPE
SOURce:DATA:TELecom:POINter:SEQuence:RATE

SOURce:DATA:TELecom:POINter:SEQuence:ANOMaly:NTIMe?

This query returns the pointer sequence time until the next anomaly in seconds.

This query is only meaningful if sequences are running (STATus? is "OPERATING").

Syntax

`SOURce:DATA:TELecom:POINter:SEQuence:ANOMaly:NTIMe?`

Response

<pointer seq ntime> (NR1-numeric)	description
Any positive integer	Time until the next anomaly in seconds
–1	Returned for sequences with an invalid NTIME (Example: continuous without anomalies) or if in the following states: STOPPED, INITIALizing, or COOLdown

Dependencies

None

Errors and Events

None

Examples

Query: `SOURCE:DATA:TELECOM:POINTER:SEQUENCE:ANOMALY:NTIME?`

Response: `4`

Related Commands

SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
SOURce:DATA:TELecom:POINter:SEQuence:TYPE
SOURce:DATA:TELecom:POINter:SEQuence:RATE

SOURce:DATA:TELecom:TRIButary Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that control the tributary signals. Figure 2–18 shows the hierarchy tree for this subsystem.

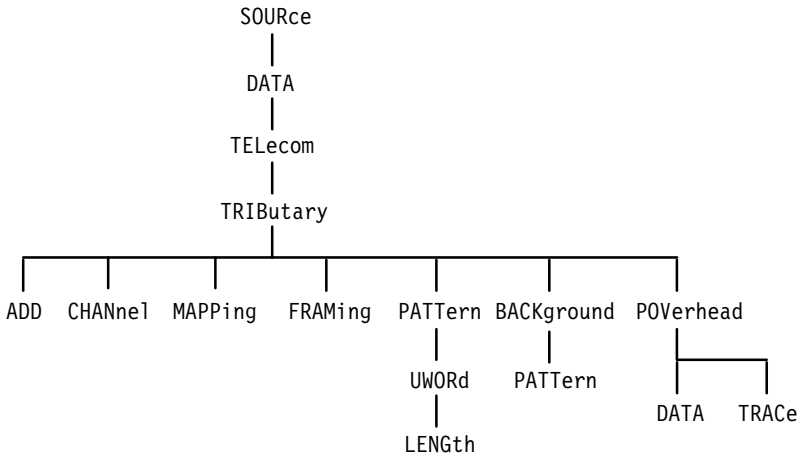


Figure 2–18: SOURce:DATA:TELecom:TRIButary Subsystem

SOURce:DATA:TELecom:TRIButary:ADD

Add/Drop/Test Option Only

This command selects the source for the tributary payload data.

Syntax `SOURce:DATA:TELecom:TRIButary:ADD <trib add>`

Parameters

<trib add> (boolean)	description
OFF or 0	Internal source; an internally generated data pattern is placed in the payload (default)
ON or 1	External tributary signal mapped into the signal; signal mapping is determined by the SOURce:DATA:TELecom:TRIButary: MAPPing command; if no signal is present, AIS is mapped into the payload

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:ADD ON`

Related Commands INPUT2 and INPUT3 subsystems

SOURce:DATA:TELecom:TRIButary:ADD?

Add/Drop/Test Option Only

This query returns the source for the tributary payload data.

Syntax `SOURce:DATA:TELecom:TRIButary:ADD?`

Response

<trib add> (boolean)	description
0	Internal source; an internally generated data pattern is placed in the payload (default)
1	External tributary signal mapped into the signal; signal mapping is determined by the SOURce:DATA:TELecom:TRIButary: MAPPing command; if no signal is present, AIS is mapped into the payload

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:ADD?`

Response: `0`

Related Commands SOURce:DATA:TELecom:TRIButary:ADD

SOURce:DATA:TELecom:TRIButary:CHANnel

Add/Drop/Test Option Only

This command selects the VTASYNC channel.

Syntax `SOURce:DATA:TELecom:TRIButary:CHANnel <trib channel>`

Parameters

<trib channel> (NR1-numeric)	description
Any integer between 1 and 28	VTASYNC (VT 1.5) mapping (default = 1)
1	DS3 mapping

Dependencies SOURce:DATA:TELecom:TRIButary:MAPPing determines the number of channels available for selection.

Errors and Events None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:CHANNEL 1`

Related Commands SOURce:DATA:TELecom:TRIButary:MAPPing

SOURce:DATA:TELecom:TRIButary:CHANnel?

Add/Drop/Test Option Only

This query returns the current VTASYNC channel.

Syntax `SOURce:DATA:TELecom:TRIButary:CHANnel?`

Response

<trib channel> (NR1-numeric)	description
Any integer between 1 and 28	VTASYNC (VT 1.5) mapping (default = 1)
1	DS3 mapping

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:CHANNEL?`

Response: 1

Related Commands SOURce:DATA:TELecom:TRIButary:CHANnel

SOURce:DATA:TELecom:TRIButary:MAPPing

Add/Drop/Test Option Only

This command selects the tributary payload mapping. When you are actively mapping and demapping a tributary signal, the SOURce:DATA:TELecom:TRIButary:MAPPing and SENSe:DATA:TELecom:TRIButary:MAPPing functions are coupled; a change to one causes the same change to the other. When this command is sent, the C2 byte of the path overhead is set.

Syntax `SOURce:DATA:TELecom:TRIButary:MAPPing <trib mapping>`

Parameters

<trib mapping> (discrete)	description
VTASYNC	Mapped DS1 signal into a VTASYNC (default)
DS3	Mapped DS3 signal

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:MAPPING DS3`

Related Commands SOURce:DATA:TELecom:PAYLoad:CHANnel

SOURce:DATA:TELecom:TRIButary:MAPPing?

Add/Drop/Test Option Only

This query returns the current tributary payload mapping.

Syntax `SOURce:DATA:TELecom:TRIButary:MAPPing?`

Response

<trib mapping> (discrete)	description
VTASYNC	Mapped DS1 signal into a VTASYNC (default)
DS3	Mapped DS3 signal

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:MAPPING?`

Response: `VTASYNC`

Related Commands SOURce:DATA:TELecom:TRIButary:MAPPing

SOURce:DATA:TELecom:TRIButary:FRAMing

Add/Drop/Test Option Only

This command selects the framing of the transmitted tributary signal.

Syntax `SOURce:DATA:TELecom:TRIButary:FRAMing <trib framing>`

Parameters

<trib framing> (discrete)	description
UNFRamed	No framing (default)
SF	DS1 superframe
ESF	DS1 extended superframe

(continued on next page)

<trib framing> (discrete)	description
CBIT	CBIT framing
M13	M13 framing

Dependencies None

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:FRAMING UNFRAMED
```

Related Commands None

SOURce:DATA:TELecom:TRIButary:FRAMing?

Add/Drop/Test Option Only

This query returns the current framing of the transmitted tributary signal.

Syntax

```
SOURce:DATA:TELecom:TRIButary:FRAMing?
```

Response

<trib framing> (discrete)	description
UNFRamed	No framing (default)
SF	DS1 superframe
ESF	DS1 extended superframe
CBIT	CBIT framing
M13	M13 framing

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:FRAMING?`

Response: `UNFRAMED`

Related Commands

SOURce:DATA:TELecom:TRIButary:FRAMing

SOURce:DATA:TELecom:TRIButary:PATTern

Add/Drop/Test Option Only

This command selects the internally generated pattern that is placed in the tributary payload.

Syntax

`SOURce:DATA:TELecom:TRIButary:PATTern <trib pattern>`

Parameters

<trib pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 is placed in the tributary payload (default)
QRSS	Quasi-random signal source pattern (DS1 rate only)
PRBS15	A pseudo-random binary sequence of length 2^{15}–1 is placed in the tributary payload
PRBS20	A pseudo-random binary sequence of length 2^{20}–1 is placed in the tributary payload
AZEROs	All zeros are placed in the payload
AONEs	All ones are placed in the payload
UWORd	A user-defined pattern is placed in the payload
FIXED_1_8	1 bit in 8 set; #H40 (DS1 rate only)
FIXED_3_24	3 bits in 24; #H440004 (DS1 rate only)

Dependencies

SOURce:DATA:TELecom:TRIButary:ADD must be set to OFF for this command to apply.

Errors and Events

None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:PATTERN PRBS15`

Related Commands SOURce:DATA:TELecom:TRIButary:PATTern:UWORd

SOURce:DATA:TELecom:TRIButary:PATTern?

Add/Drop/Test Option Only

This query returns the current internally generated tributary payload pattern.

Syntax `SOURce:DATA:TELecom:TRIButary:PATTern?`

Response

<trib pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 is in the tributary payload (default)
QRSS	Quasi-random signal source pattern (DS1 rate only)
PRBS15	A pseudo-random binary sequence of length 2^{15}–1 is in the tributary payload
PRBS20	A pseudo-random binary sequence of length 2^{20}–1 is in the tributary payload
AZEROs	All zeros are in the payload
AONEs	All ones are in the payload
UWORd	A user-defined pattern is in the payload
FIXED_1_8	1 bit in 8 set; #H40 (DS1 rate only)
FIXED_3_24	3 bits in 24; #H440004 (DS1 rate only)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:PATTERN?`

Response: `PRBS23`

Related Commands SOURce:DATA:TELecom:TRIButary:PATTern

SOURce:DATA:TELecom:TRIButary:PATTern:UWORd

Add/Drop/Test Option Only

This command sets the user-defined pattern that is placed in the tributary payload.

Syntax `SOURce:DATA:TELecom:TRIButary:PATTern:UWORd <trib user pattern>`

Parameters

<trib user pattern> (hexadecimal)	description
Any 8, 16, or 24 bit hexadecimal number in the range #H00 to #HFFFFFF	Repeating pattern is placed in the tributary payload (default = #H00)

Dependencies SOURce:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this command to apply. Use the SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth command to set the length of the repeating pattern.

Errors and Events None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD #HAA5500`

Related Commands SOURce:DATA:TELecom:TRIButary:PATTern
SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth

SOURce:DATA:TELecom:TRIButary:PATTern:UWORd?

Add/Drop/Test Option Only

This query returns the user-defined pattern that is placed in the tributary payload.

Syntax `SOURce:DATA:TELecom:TRIButary:PATTern:UWORd?`

Response

<trib user pattern> (hexadecimal)	description
Any 8, 16, or 24 bit hexadecimal number in the range #H00 to #HFFFFFF	Repeating pattern is placed in the tributary payload (default = #H00)

Dependencies SOURce:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this query to apply. Use the SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth command to set the length of the repeating pattern.

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD?`

Response: `#HAA5500`

Related Commands SOURce:DATA:TELecom:TRIButary:PATTern:UWORd

SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENGth

Add/Drop/Test Option Only

This command sets the number of bytes of the user-defined pattern that are repeated in the tributary payload.

Syntax `SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENGth <trib user pattern length>`

Parameters

<trib user pattern length> (NR1-numeric)	description
Any integer in the range 1 to 3	Number of bytes of user-defined pattern that are repeated in the tributary payload (default = 1)

Dependencies SOURce:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this command to apply. Use the SOURce:DATA:TELecom:TRIButary:PATTern:UWORd command to set the repeating pattern.

Errors and Events None

Examples SOURCE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD:LENGTH 3

Related Commands SOURce:DATA:TELecom:TRIButary:PATTern:UWORd

SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENGth?

Add/Drop/Test Option Only

This query returns the number of bytes of the user-defined pattern that are repeated in the tributary payload.

Syntax SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENGth?

Response

<trib user pattern length> (NR1-numeric)	description
Any integer in the range 1 to 3	Number of bytes of user-defined pattern that are repeated in the tributary payload (default = 1)

Dependencies SOURce:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this query to apply. Use the SOURce:DATA:TELecom:TRIButary:PATTern: UWORd:LENGth command to set the length of the repeating pattern.

Errors and Events None

Examples Query: SOURCE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD:LENGTH?

Response: 3

Related Commands SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENGth

SOURce:DATA:TELecom:TRIButary:BACKground:PATTern

Add/Drop/Test Option Only

This command selects the internally generated pattern that is placed in the tributary payload for inactive channels.

Syntax SOURce:DATA:TELecom:TRIButary:BACKground:PATTern <trib background pattern>

Parameters

<trib background pattern> (discrete)	description
QRSS	Quasi-random signal source pattern (VTA-SYNC only) (default)
IDLE	An idle pattern is placed in the tributary payload of inactive channels

Dependencies None

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:BACKGROUND:PATTERN IDLE
```

Related Commands None

SOURce:DATA:TELecom:TRIButary:BACKground:PATTern?

Add/Drop/Test Option Only

This query returns the current internally generated tributary payload pattern placed in inactive channels.

Syntax

```
SOURce:DATA:TELecom:TRIButary:BACKground:PATTern?
```

Response

<trib background pattern> (discrete)	description
QRSS	Quasi-random signal source pattern (VTA-SYNC only) (default)
IDLE	An idle pattern is placed in the tributary payload of inactive channels

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:BACKGROUND:PATTERN?`

Response: `IDLE`

Related Commands SOURce:DATA:TELecom:TRIButary:BACKground:PATTern

SOURce:DATA:TELecom:TRIButary:POVerhead:DATA

This command sets the bytes in the tributary path overhead.

Syntax `SOURce:DATA:TELecom:TRIButary:POVerhead:DATA <byte name>,<value>`

Parameters

<byte name> (discrete)	description
V5	(VT1.5)
<value> (NR1-numeric)[1]	**description**
Any integer in the range 0 to 255 (hexadecimal 00 to FF)	The selected byte is set to this value

1 **A hexadecimal value is also acceptable.**

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:POVERHEAD:DATA V5,#H55`

Related Commands SOURce:DATA:TELecom:TRIButary:POVerhead:TRACe

SOURce:DATA:TELecom:TRIButary:POVerhead:DATA?

This query returns the value in the specified tributary path overhead bytes. (Applies to VT1.5 mapping.)

Syntax `SOURce:DATA:TELecom:TRIButary:POVerhead:DATA? <byte>`

Parameters

<byte> (discrete)	description
V5	(VT1.5)

Response

<value> (NR1-numeric)	description
Any integer in the range 0 to 255	The selected byte is set to this value

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POVERHEAD:DATA? V5`

Response: `255`

Related Commands SOURce:DATA:TELecom:TRIButary:POVerhead:DATA

SOURce:DATA:TELecom:TRIButary:ERRor, ALARm, FAILure Subsystem

This section describes subsystem commands that control abnormal conditions in the transmitted tributary signal.

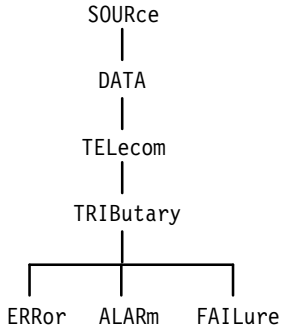


Figure 2–19: SOURce:DATA:TELecom:TRIButary:ERRor, ALARm, FAILure Subsystem

SOURce:DATA:TELecom:TRIButary:ERRor

Add/Drop/Test Option Only

This command selects the type of tributary error that is transmitted at a rate specified by SOURce:DATA:TELecom:ERRor:RATE. Use the SOURce:DATA:TELecom:ERRor:IMMediate command to insert the error.

Syntax `SOURce:DATA:TELecom:TRIButary:ERRor <trib error>`

Parameters

<trib error> (discrete)	description
NONE	No error transmitted
DATA	Error in pattern
FRAMe	Frame error
CRC	CRC error; you must be transmitting a DS1 tributary signal and framing must be set to ESF
PARITY	Parity error; you must be transmitting a DS3 tributary signal
VTFEbe	VT FEBE
VTBIP	VT BIP

Dependencies SOURce:DATA:TELecom:ERRor:TYPE must be set to TRIButary for this query to apply.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:ERROR DATA
```

Related Commands SOURce:DATA:TELecom:ERRor:TYPE
SOURce:DATA:TELecom:ERRor:RATE
SOURce:DATA:TELecom:ERRor:IMMediate

SOURce:DATA:TELecom:TRIButary:ERRor?

Add/Drop/Test Option Only

This query returns the current setting of the tributary error type.

Syntax

```
SOURce:DATA:TELecom:TRIButary:ERRor?
```

Response

<trib error> (discrete)	description
NONE	No error transmitted
DATA	Error in pattern
FRAMe	Frame error
CRC	CRC error; you must be transmitting a DS1 tributary signal and framing must be set to ESF
PARITY	Parity error; you must be transmitting a DS3 tributary signal
VTFEbe	VT FEBE
VTBIP	VT BIP

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIB:ERROR?`

Response: `DATA`

Related Commands SOURce:DATA:TELecom:TRIButary:ERRor

SOURce:DATA:TELecom:TRIButary:ALARm

Add/Drop/Test Option Only

This command selects a tributary alarm to transmit.

Syntax `SOURce:DATA:TELecom:TRIButary:ALARm <trib alarm>`

Parameters

<trib alarm> (discrete)	description
NONE	No alarm transmitted (default)
VTPAIS	VT path AIS
VTFERF	VT FERF
AIS	Tributary AIS
YELlow	Tributary yellow
IDLE	DS3 idle

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:ALARM NONE`

Related Commands None

SOURce:DATA:TELecom:TRIButary:ALARm?

Add/Drop/Test Option Only

This query returns the current setting of the transmitted tributary alarm type.

Syntax `SOURce:DATA:TELecom:TRIButary:ALARm?`

Response

<trib alarm> (discrete)	description
NONE	No alarm transmitted (default)
VTPAIS	VT path AIS
VTFERF	VT FERF
AIS	Tributary AIS
YELlow	Tributary yellow
IDLE	DS3 idle

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIB:ALARM?`

Response: `NONE`

Related Commands SOURce:DATA:TELecom:TRIButary:ALARm

SOURce:DATA:TELecom:TRIButary:FAILure

Add/Drop/Test Option Only

This command selects a tributary failure to transmit.

Syntax `SOURce:DATA:TELecom:TRIButary:FAILure <trib failure>`

Parameters

<trib failure> (discrete)	description
NONE	No failure transmitted (default)
VTLOP	VT Loss of Pointer
VTLOM	VT Loss of Multiframe

Dependencies None

Errors and Events None

Examples `SOURCE:DATA:TELECOM:TRIBUTARY:FAILURE NONE`

Related Commands None

SOURce:DATA:TELecom:TRIButary:FAILure?

Add/Drop/Test Option Only

This query returns the current setting of the transmitted tributary failure type.

Syntax `SOURce:DATA:TELecom:TRIButary:FAILure?`

Response

<trib failure> (discrete)	description
NONE	No failure transmitted (default)
VTLOP	VT Loss of Pointer
VTLOM	VT Loss of Multiframe

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIB:FAILURE?`

Response: `NONE`

Related Commands SOURce:DATA:TELecom:TRIButary:FAILure

SOURce:DATA:TELecom:TRIButary:POINter Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that adjust pointers for the transmitted VT mapped signal. Figure 2–20 shows the hierarchy tree for this subsystem.

***NOTE**. SOURce:DATA:TELecom:POINter:MODE must be set to TRIButary for any command or query in this section to apply.*

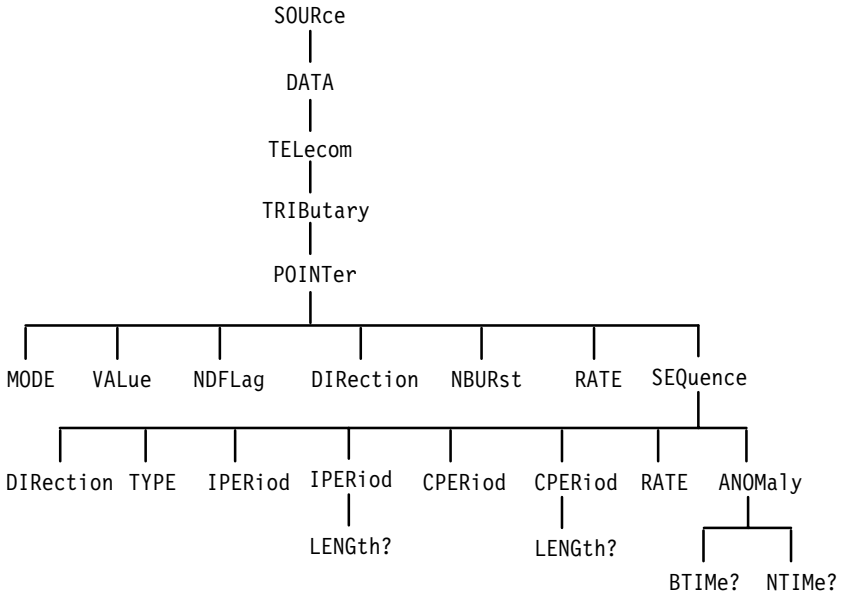


Figure 2–20: SOURce:DATA:TELecom:TRIButary:POINTer Subsystem

SOURce:DATA:TELecom:TRIButary:POINter:MODE

Add/Drop/Test Option Only

This command controls the VT pointer manipulation modes.

Syntax `SOURce:DATA:TELecom:TRIButary:POINter:MODE <trib pointer mode>`

Parameters

<trib pointer mode> (discrete)	description
MANual	Pointers are controlled by SOURce:DATA:TRIButary:POINter:VALue and SOURce:DATA:TRIButary:POINter:NDFLag (default)
SINGle	Pointer adjustments are controlled by the SOURce:DATA:TELecom:POINter:ACTion and SOURce:DATA:TELecom:POINter:DIRection commands
BURSt	When the SOURce:DATA:TELecom:POINter:ACTion command is given, a burst of pointer adjustments is sent at the maximum rate (1 in 4 frames) with a count defined by SOURce:DATA:TRIButary:POINter:NBURst
FOFFset	Tributary pointers are controlled by the SOURce:CLOCk:OFFSet subsystem.
CONTinuous	Pointers are continuously adjusted according to the SOURce:DATA:TRIButary:POINter:DIRection and SOURce:DATA:TRIButary:POINter:RATE commands
SEQuence	Pointers are stressed according to sequences defined in T1.105.03–1994 or G.783

Dependencies SOURce:DATA:TELecom:POINter:MODE must be set to TRIButary for this command to apply. You can control only STS or VT pointer adjustments at any one time.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:MODE MANUAL
```

Related Commands SOURce:DATA:TELecom:POINter:MODE

SOURce:DATA:TELecom:TRIButary:POINter:MODE?

Add/Drop/Test Option Only

This query returns the current setting of the VT pointer mode.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:MODE?
```

Response

<trib pointer mode> (discrete)	description
MANual	Pointers are controlled by SOURce:DATA:TRIButary:POINter:VALue and SOURce:DATA:TRIButary:POINter:NDFLag (default)
SINGle	Pointer adjustments are controlled by the SOURce:DATA:TELecom:POINter:ACTion and SOURce:DATA:TELecom:POINter:DIRection commands
BURSt	When the SOURce:DATA:TELecom:POINter:ACTion command is given, a burst of pointer adjustments is sent at the maximum rate (1 in 4 frames) with a count defined by SOURce:DATA:TRIButary:POINter:NBURst
FOFFset	Tributary pointers are controlled by the SOURce:CLOCk:OFFSet subsystem.
CONTinuous	Pointers are continuously adjusted according to the SOURce:DATA:TRIButary:POINter:DIRection and SOURce:DATA:TRIButary:POINter:RATE commands
SEQuence	Pointers are stressed according to sequences defined in T1.105.03–1994 or G.783

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:MODE?`

Response: `MANUAL`

Related Commands SOURce:DATA:TELecom:TRIButary:POINter:MODE

SOURce:DATA:TELecom:TRIButary:POINter:VALue

Add/Drop/Test Option Only

This command sets the VT pointer value. If SOURce:DATA:TELecom:TRIButary:POINter:NDFLag is set to ON, a New Data Flag (NDF) is sent with each new value received.

Syntax SOURce:DATA:TELecom:TRIButary:POINter:VALue <trib pointer value>

Parameters

<trib pointer value> (NR1-numeric)	description
Any integer in the range 0 to 1023	VTASYNC (default = 78, illegal > 103)

Dependencies SOURce:DATA:TELecom:POINter:MODE must be set to TRIButary for this command to apply. You can control only STS or VT pointer adjustments at any one time.

SOURce:DATA:TELecom:TRIButary:POINter:MODE must be set to MANual for this command to apply.

Errors and Events None

Examples SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:VALUE 10

Related Commands SOURce:DATA:TELecom:TRIButary:POINter:MODE

SOURce:DATA:TELecom:TRIButary:POINter:VALue?

Add/Drop/Test Option Only

This query returns the current VT pointer value transmitted.

Syntax SOURce:DATA:TELecom:TRIButary:POINter:VALue?

Response

<trib pointer value> (NR1-numeric)	description
Any integer in the range 0 to 1023	VTASYNC (default = 78, illegal > 103)

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:VALUE?`

Response: `102`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:VALue

SOURce:DATA:TELecom:TRIButary:POINter:NDFLag

Add/Drop/Test Option Only

This command controls the generation of a New Data Flag (NDF) when VT pointer adjustments occur.

Syntax

`SOURce:DATA:TELecom:TRIButary:POINter:NDFLag <trib NDF state>`

Parameters

<trib NDF state> (boolean)	description
1 or ON	On (default)
0 or OFF	Off

Dependencies

SOURce:DATA:TELecom:POINter:MODE must be set to TRIButary for this command to apply. You can control only STS or VT pointer adjustments at any one time.

SOURce:DATA:TELecom:TRIButary:POINter must be set to MANual for this command to apply.

Errors and Events

None

Examples

`SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:NDFLAG ON`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:VALue
SOURce:DATA:TELecom:TRIButary:POINter:MODE

SOURce:DATA:TELecom:TRIButary:POINter:NDFLag?

Add/Drop/Test Option Only

This query returns the current setting of the VT New Data Flag (NDF) generator.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:NDFLag?
```

Response

<trib NDF state> (boolean)	description
1	On (default)
0	Off

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:NDFLAG?`

Response: 0

Related Commands SOURce:DATA:TELecom:TRIButary:POINter:NDFLag

SOURce:DATA:TELecom:TRIButary:POINter:DIRection

Add/Drop/Test Option Only

This command sets the direction of continuous VT pointer adjustments.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:DIRection <trib pointer direction>
```

Parameters

<trib pointer direction> (discrete)	description
ALTernate	Pointer adjustments alternate between up and down (default)
DOWN	Pointers adjusted down
UP	Pointers adjusted up

Dependencies SOURce:DATA:TELecom:POINter:MODE must be set to TRIButary for this command to apply. You can control only STS or VT pointer adjustments at any one time.

SOURce:DATA:TELecom:TRIButary:POINTer:MODE must be set to CONTinuous for this command to apply.

Errors and Events None

Examples
```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:DIRECTION UP
```

Related Commands SOURce:DATA:TELecom:TRIButary:POINter:MODE
SOURce:DATA:TELecom:TRIButary:POINter:RATE

SOURce:DATA:TELecom:TRIButary:POINter:DIRection?

Add/Drop/Test Option Only

This query returns the current setting for the direction of continuous VT pointer adjustments.

Syntax
```
SOURce:DATA:TELecom:TRIButary:POINter:DIRection?
```

Response

<trib pointer direction> (discrete)	description
ALTernate	Pointer adjustments alternate between up and down (default)
DOWN	Pointers adjusted down
UP	Pointers adjusted up

Dependencies None

Errors and Events None

Examples Query:
```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:DIRECTION?
```
Response:
```
ALTERNATE
```

Related Commands SOURce:DATA:TELecom:TRIButary:POINter:DIRection

SOURce:DATA:TELecom:TRIButary:POINter:NBURst

Add/Drop/Test Option Only

This command sets the number of pointer adjustments in a burst of VT pointer adjustments. Send the SOURce:DATA:TELecom:POINter:ACTion command to create the burst of pointer adjustments.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:NBURst <trib pointer burst>
```

Parameters

<trib pointer burst> (NR1-numeric)	description
Any integer in the range 2 to 8	This value determines the number of pointer adjustments in a burst of pointer adjustments (default = 2)

Dependencies

SOURce:DATA:TELecom:POINter:MODE must be set to TRIButary for this command to apply. You can control only STS or VT pointer adjustments at any one time.

SOURce:DATA:TELecom:TRIButary:POINter:MODE must be set to BURSt for this command to apply.

Errors and Events

None

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:NBURST 2
```

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:MODE
SOURce:DATA:TELecom:POINter:ACTion
SOURce:DATA:TELecom:POINter:MODE

SOURce:DATA:TELecom:TRIButary:POINter:NBURst?

Add/Drop/Test Option Only

This query returns the number of pointer adjustments in a burst of VT pointer adjustments.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:NBURst?
```

Response

<trib pointer burst> (NR1-numeric)	description
Any integer in the range 2 to 8	This value determines the number of pointer adjustments in a burst of pointer adjustments (default = 2)

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:NBURST?`

Response: 3

Related Commands SOURce:DATA:TELecom:TRIButary:POINter:NBURst

SOURce:DATA:TELecom:TRIButary:POINter:RATE

Add/Drop/Test Option Only

This command sets the continuous VT pointer adjustment rate.

Syntax `SOURce:DATA:TELecom:TRIButary:POINter:RATE <trib pointer rate>`

Parameters

<trib pointer rate> (NR1-numeric)	description
Any integer in the range 2 to 10,000 ms (resolution of 1 ms)	The pointer adjustment rate is set to this value

Dependencies SOURce:DATA:TELecom:POINter:MODE must be set to TRIButary for this command to apply. You can control only STS or VT pointer adjustments at any one time.

SOURce:DATA:TELecom:TRIButary:POINter:MODE must be set to CONTinuous for this command to apply.

Errors and Events None

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:RATE 9
```

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:DIRection
SOURce:DATA:TELecom:TRIButary:POINter:MODE
SOURce:DATA:TELecom:POINter:MODE

SOURce:DATA:TELecom:TRIButary:POINter:RATE?

Add/Drop/Test Option Only

This query returns the current setting of the VT pointer adjustment rate.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:RATE?
```

Response

<trib pointer rate> (NR1-numeric)	description
Any integer in the range 2 to 10,000 ms (resolution of 1 ms)	The pointer adjustment rate is set to this value

Dependencies

None

Errors and Events

None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:RATE?`

Response: 3

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:RATE

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:DIRection

Add/Drop/Test Option Only

This command sets the pointer movement direction.

Setting this parameter when a sequence is running returns an error.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:DIRection
<pointer seq direction>
```

Parameters

<pointer seq direction> (discrete)	description
DOWN	(default)
UP	

Dependencies Pointer sequences must not be running.

Errors and Events 221, “Settings conflict; stop sequences before setting the direction”

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQuence:DIRECTION UP
SOUR:DATA:TEL:TRIB:POIN:SEQ:DIR DOWN
```

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:DIRection?

Add/Drop/Test Option Only

This query returns the direction for sequence pointer movements.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:DIRection?
```

Response

<pointer seq direction> (discrete)	description
DOWN	(default)
UP	

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:DIRECTION?`

Response: `UP`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:DIRection

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE

Add/Drop/Test Option Only

This command sets the pointer sequence movement rate in milliseconds.

Setting this parameter when a sequence is running returns an error.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE
<pointer seq rate>
```

Parameters

<pointer seq rate> (NR1-numeric)	description
Any integer from 34 to 30,000 ms	Resolution is 1 ms (default depends on mapping and type)

Dependencies

Pointer sequences must not be running. Some sequence types (like single, burst, phase, sinalt, and dblalt) set the rate to 30,000 ms. With these types, you cannot change the rate.

Errors and Events

221, “Settings conflict; stop sequences before setting the rate”

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:RATE 2000
SOUR:DATA:TEL:TRIB:POIN:SEQ:RATE 1000
```

Related Commands

SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
SOURce:DATA:TELecom:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE?

Add/Drop/Test Option Only

This query returns the pointer movement rate in milliseconds.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE?
```

Response

<pointer seq rate> (NR1-numeric)	description
Any integer from 34 to 30,000 ms	Resolution is 1 ms (default depends on mapping and type)

Dependencies

Some sequence types (like single, burst, phase, sinalt, and dblalt) set the rate to 30,000 ms. With these types, you cannot change the rate.

Errors and Events

None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:RATE?`

Response: `2000`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE

Add/Drop/Test Option Only

This command sets the pointer sequence type.

Setting this parameter when a sequence is running returns an error.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE
<pointer seq type>
```

Parameters

<pointer seq type> (discrete)	description	standard
SINGle	Single pointer adjustment	ANSI
BURSt	Burst pointer adjustment	ANSI

(continued on next page)

<pointer seq type> (discrete)	description	standard
PHASe	Phase transient pointer adjustment	ANSI
P261	Periodic 26–1 pointer adjustment	ANSI
P261CAN	Periodic 26–1 with cancel	ANSI
P261ADD	Periodic 26–1 with add	ANSI
PCONtinuous	Periodic continuous pointer adjustment	ANSI
PCONCAN	Periodic continuous with cancel	ANSI
PCONADD	Periodic continuous with add	ANSI
SINALT	Single alternating pointer	ITU-T
DBLALT	Double alternating pointer	ITU-T

Dependencies Pointer sequences must not be running.

Errors and Events 221, “Settings conflict; stop sequences before setting type”

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:TYPE BURST
SOUR:DATA:TEL:TRIB:POIN:SEQ:TYPE BURS
```

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE?

Add/Drop/Test Option Only

This query returns the pointer sequence type.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE?
```

Response

<pointer seq type> (discrete)	description
SINGle	Single pointer adjustment
BURSt	Burst pointer adjustment

(continued on next page)

<pointer seq type> (discrete)	description
PHASe	Phase transient pointer adjustment
P261	Periodic 26–1 pointer adjustment
P261CAN	Periodic 26–1 with cancel
P261ADD	Periodic 26–1 with add
PCONtinuous	Periodic continuous pointer adjustment
PCONCAN	Periodic continuous with cancel
PCONADD	Periodic continuous with add
SINALT	Single alternating pointer
DBLALT	Double alternating pointer

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:TYPE?`

Response: `PCONCAN`

Related Commands SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod

Add/Drop/Test Option Only

This command enables or disables the pointer sequence initialization period.

Setting this parameter when a sequence is running returns an error.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod
<pointer seq init>
```

Parameters

<pointer seq init> (discrete)	description
0 (or OFF)	Disables the pointer sequence initialization period
1 (or ON)	Enables the pointer sequence initialization period (default)

Dependencies Pointer sequences must not be running.

Errors and Events 221, “Settings conflict; stop sequences before setting the period”

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:IPERIOD ON
```

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod?

Add/Drop/Test Option Only

This query returns the pointer sequence initialization period enable/disable condition.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod?
```

Response

<pointer seq init> (discrete)	description
0	Disabled
1	Enabled (default)

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:IPERIOD?`

Response: `0`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod:LENGth?

Add/Drop/Test Option Only

This query returns the pointer sequence initialization period in seconds.

Syntax

`SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod:LENGth?`

Response

<pointer seq init> (NR1-numeric)	description
Any integer	Pointer sequence initialization period in seconds

Dependencies

None

Errors and Events

None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:IPER-IOD:LENGTH`

Response: `30`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod

Add/Drop/Test Option Only

This command enables or disables the pointer sequence cool down period.

Setting this parameter when a sequence is running returns an error.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod
<pointer seq init>
```

Parameters

<pointer seq init> (discrete)	description
0 (or OFF)	Disables the pointer sequence cool down period
1 (or ON)	Enables the pointer sequence cool down period (default)

Dependencies Pointer sequences must not be running.

Errors and Events 221, “Settings conflict; stop sequences before setting the period”

Examples

```
SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:CPERIOD OFF
```

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod?

Add/Drop/Test Option Only

This query returns the pointer sequence cool down period enable/disable condition.

Syntax

```
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod?
```

Response

<pointer seq init> (discrete)	description
0	Disabled
1	Enabled (default)

Dependencies None

Errors and Events None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:CPERIOD?`

Response: `0`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod:LENGth?

Add/Drop/Test Option Only

This query returns the pointer sequence cool down period in seconds.

Syntax

`SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod:LENGth?`

Response

<pointer seq init> (NR1-numeric)	description
Any integer from 900 to 60	Pointer sequence cool down period in seconds

Dependencies

None

Errors and Events

None

Examples

Query: `SOURCE:DATA:TELECOM:TRIBUTARY:POINTER:SEQUENCE:CPERIOD:LENGTH`

Response: `235`

Related Commands

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:ANOMaly:BTIMe?

This query returns the pointer sequence time between anomalies in seconds.

Sequences do not have to be running. This calculation is based upon sequence type and rate.

Syntax

`SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:ANOMaly:BTIMe?`

Response

<pointer seq btime> (NR1-numeric)	description
Any positive integer	Time between anomalies in seconds
–1	Returned for sequences with an invalid BTIME. Example: continuous without anomalies

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIButary:POINTER:SEQUENCE:ANOMALY:BTIME?`

Response: `30`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE

SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:ANOMaly:NTIMe?

This query returns the pointer sequence time until the next anomaly in seconds.

This query is only meaningful if sequences are running.

Syntax `SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:ANOMaly:NTIMe?`

Response

<pointer seq ntime> (NR1-numeric)	description
Any positive integer	Time until the next anomaly in seconds
–1	Returned for sequences with an invalid NTIME (Example: continuous without anomalies) or if in the following states: STOPPED, INITIALizing, or COOLdown

Dependencies None

Errors and Events None

Examples Query: `SOURCE:DATA:TELECOM:TRIButary:POINTER:SEQUENCE:ANOMALY:NTIME?`

Response: `4`

Related Commands SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:SEQuence:CONTrol
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE

Receive Commands

The Receive Commands allow you to set up the physical connections of a received signal and measure the signal. This section contains all of the commands and queries for each of the following Receive subsystems:

- INPUT1
- INPUT2 (Add/Drop/Test Option Only)
- INPUT3 (Add/Drop/Test Option Only)
- SENSe:DATA:TELecom
- SENSe:DATA:TELecom:TEST
- SENSe:DATA:TELecom:OVERhead and POVerhead
- SENSe:DATA:TELecom:MEASure
- SENSe:DATA:TELecom:MEASure:STESts
- SENSe:DATA:TELecom:AUTOscan
- SENSe:DATA:TELecom:TRIButary (Add/Drop/Test Option Only)
- SENSe:DATA:TELecom:MEASure:TRIButary (Add/Drop/Test Option Only)

INPUT1 Subsystem

This section describes the commands and queries that allow you to set the rate, type, and level of the incoming signal. Figure 2–21 shows the hierarchy tree for this subsystem.

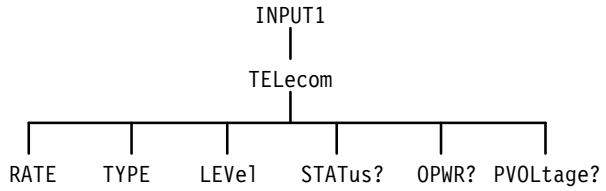


Figure 2–21: INPUT1 Subsystem

INPUT1:TELecom:RATE

This command selects the signal rate of the receiver. The signal connected to INPUT1 is passed to the receiver by the SENSe:DATA:TELecom:SOURce INPUT1 command.

Syntax

```
INPUT1:TELecom:RATE <rate>
```

Parameters

<rate> (discrete)	description
STS1	51.84 MHz (default)
STS3	155.52 MHz
STS12	622.08 MHz

Dependencies

None

Errors and Events

221, “Settings conflict; Rate is not available with current Line Interface module or operating mode”

Examples

```
INPUT1:TELECOM:RATE STS1
```

Related Commands

SENSe:DATA:TELecom:SOURce

INPUT1:TELecom:RATE?

This query returns the current setting of the input signal rate.

Syntax `INPUT1:TELecom:RATE?`

Response

<rate> (discrete)	description
STS1	51.84 MHz (default)
STS3	155.52 MHz
STS12	622.08 MHz

Dependencies None

Errors and Events None

Examples Query: `INPUT1:TELecom:RATE?`

Response: `STS1`

Related Commands INPUT1:TELecom:RATE

INPUT1:TELecom:TYPE

This command selects the input signal type. The Operation Complete bit in the Standard Event Status Register is set to 1 when this command is executed.

Syntax `INPUT1:TELecom:TYPE <type>`

Parameters

<type> (discrete)	description
ELECtrical	Electrical output (default)
OPTical	Optical output

Dependencies None

Errors and Events 221, “Settings conflict; Type is not available with current Line Interface module”

Examples `INPUT1:TELECOM:TYPE ELECTRICAL`

Related Commands None

INPUT1:TELecom:TYPE?

This query returns the selected input signal type.

Syntax `INPUT1:TELecom:TYPE?`

Response

<type> (discrete)	description
ELECtrical	Electrical signal input (default)
OPTical	Optical signal input

Dependencies None

Errors and Events None

Examples Query: `INPUT1:TELECOM:TYPE?`

Response: `OPTICAL`

Related Commands INPUT1:TELecom:TYPE

INPUT1:TELecom:LEVel

This command selects the expected level of the received electrical signal.

Some optical and electrical modules do not support all levels. Check the status event register to verify valid settings.

Syntax `INPUT1:TELecom:LEVel <level>`

Parameters

<level> (discrete)	description
XCONnect	Simulates cross connect level (default)
LOW	Input level is low (for STS-1 rate)

<level> (discrete)	description
HIGH	High level
MONitor	Monitor level

Dependencies INPUT1:TELecom:TYPE must be set to ELECtrical for this command to apply.

Errors and Events 221, “Settings conflict; Argument is not valid in current instrument state”

Examples

```
INPUT1:TELECOM:LEVEL XCON
```

Related Commands INPUT1:TELecom:TYPE

INPUT1:TELecom:LEVel?

This query returns the expected level of the received electrical signal.

Some optical and electrical modules do not support all levels. Check the status event register to verify valid settings.

Syntax

```
INPUT1:TELecom:LEVel?
```

Response

<level> (discrete)	description
XCONnect	Simulates cross connect level (default)
LOW	Input level is low (for STS-1 rate)
HIGH	High level
MONitor	Monitor level

Dependencies None

Errors and Events None

Examples Query:

```
INPUT1:TELECOM:LEVEL?
```

Response:

```
XCONNECT
```

Related Commands INPUT1:TELecom:LEVel

INPUT1:TELecom:STATus?

This query returns the status of the received signal connected to INPUT1. The response is valid only when the receiver is set up to receive SONET signals.

Syntax `INPUT1:TELecom:STATus?`

Response

<status> (discrete)	description
NORMal	Normal signal received
MONitor	Monitor point signal received (electrical only)
LOSignal	No signal received

Dependencies None

Errors and Events None

Examples Query: `INPUT1:TELECOM:STATUS?`

Response: `MONITOR`

Related Commands INPUT1:TELecom:LEVel

INPUT1:TELecom:OPWR?

This query returns the optical signal level in dBm.

Syntax `INPUT1:TELecom:OPWR?`

Response

<optical level> (NR3-numeric)	description
Any floating point number	The optical signal level of the received signal in dBm

Dependencies INPUT1:TELecom:TYPE must be set to OPTical.

Errors and Events None

Examples Query: `INPUT1:TELECOM:OPWR?`

Response: `-25.0`

Related Commands INPUT1:TELecom:TYPE

INPUT1:TELecom:PVOLtage?

This query returns the peak electrical voltage in volts.

Syntax `INPUT1:TELecom:PVOLtage?`

Response

<peak voltage> (NR3-numeric)	description
Any floating point number	The peak voltage of the received signal in volts

Dependencies INPUT1:TELecom:TYPE must be set to ELECtrical.

Errors and Events None

Examples Query: `INPUT1:TELECOM:PVOLTAGE?`

Response: `0.52`

Related Commands INPUT1:TELecom:TYPE

INPUT2 Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that set the characteristics of the received or added DS1 tributary signal.

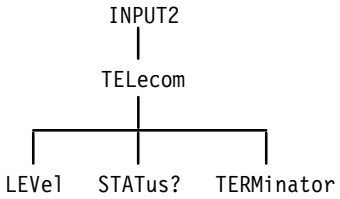


Figure 2–22: INPUT2 Subsystem

INPUT2:TELecom:LEVel

Add/Drop/Test Option Only

This command selects the expected signal level at the DS1 receive connector.

Syntax `INPUT2:TELecom:LEVel <trib1 input level>`

Parameters

<trib1 input level> (discrete)	description
NORMal	Normal input levels (default)
MONitor	Monitor level
BRIDge	Bridged input

Dependencies The instrument must be set up to receive or add a DS1 tributary signal for this command to apply.

Errors and Events None

Examples `INPUT2:TELECOM:LEVEL NORMAL`

Related Commands SENSe:DATA:TELecom:SOURce
SOURce:DATA:TELecom:TRIButary:ADD

INPUT2:TELecom:LEVel?

Add/Drop/Test Option Only

This query returns the expected signal level at the DS1 receive connector.

Syntax `INPUT2:TELecom:LEVel?`

Response

<trib1 input level> (discrete)	description
NORMal	Normal input levels (default)
MONitor	Monitor level
BRIDge	Bridged input

Dependencies None

Errors and Events None

Examples Query: `INPUT2:TELECOM:LEVEL?`

Response: `NORMAL`

Related Commands INPUT2:TELecom:LEVel

INPUT2:TELecom:STATus?

Add/Drop/Test Option Only

This query returns the status of the received or added DS1 tributary signal.

Syntax `INPUT2:TELecom:STATus?`

Response

<trib1 input status> (discrete)	description
NORMal	Signal is of acceptable quality
LOSignal	Loss of Signal (no signal connected)

Dependencies SENSe:DATA:TELecom:SOURce must be set to INPUT2 for this command to apply.

Errors and Events None

Examples Query: `INPUT2:TELECOM:STATUS?`

Response: `NORMAL`

Related Commands SENSe:DATA:TELecom:SOURce

INPUT2:TELecom:TERMinator

Add/Drop/Test Option Only

This command selects the signal terminator for the DS1 receive connector.

Syntax `INPUT2:TELecom:TERMinator <trib1 input termin>`

Parameters

<trib1 input termin> (discrete)	description
BALanced	120 Ω connector (default)

Dependencies The instrument must be set up to receive or add a DS1 tributary signal for this command to apply.

Errors and Events None

Examples `INPUT2:TELECOM:TERMINATOR BALANCED`

Related Commands SENSe:DATA:TELecom:SENSe
SOURce:DATA:TELecom:TRIButary:ADD

INPUT2:TELecom:TERMinator?

Add/Drop/Test Option Only

This returns the current setting of the DS1 transmit connector signal terminator.

Syntax `INPUT2:TELecom:TERMinator?`

Response

<trib1 input termin> (discrete)	description
BALanced	120 Ω connector (default)

Dependencies None

Errors and Events None

Examples Query: `INPUT2:TELECOM:TERMINATOR?`

Response: `BALANCED`

Related Commands INPUT2:TELecom:TERMinator

INPUT3 Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that set the characteristics of the received or added DS3 tributary signal.

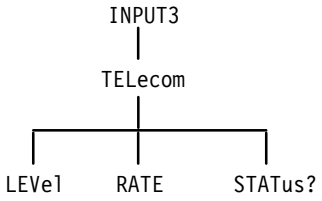


Figure 2–23: INPUT3 Subsystems

INPUT3:TELecom:LEVel

Add/Drop/Test Option Only

This command selects the expected signal level at the DS3 receive connector.

Syntax `INPUT3:TELecom:LEVel <trib2 input level>`

Parameters

<trib2 input level> (discrete)	description
NORMal	Normal input level (default)
MONitor	Monitor level

Dependencies The instrument must be set up to receive or add a DS3 tributary signal for this command to apply.

Errors and Events None

Examples `INPUT3:TELECOM:LEVEL NORMAL`

Related Commands None

INPUT3:TELecom:LEVel?

Add/Drop/Test Option Only

This query returns the expected signal level at the DS3 receive connector.

Syntax `INPUT3:TELecom:LEVel?`

Response

<trib2 input level> (discrete)	description
NORMal	Normal input level
MONitor	Monitor level

Dependencies None

Errors and Events None

Examples Query: `INPUT3:TELECOM:LEVEL?`

Response: `NORMAL`

Related Commands INPUT3:TELecom:LEVel

INPUT3:TELecom:RATE

Add/Drop/Test Option Only

This command selects the DS3 or DS3_DEMUX tributary input rate.

Syntax `INPUT3:TELecom:RATE <trib2 input rate>`

Parameters

<trib2 input rate> (discrete)	description
DS3	44.736 Mb/s (default)
DS3_DEMUX	44.736 Mb/s DS3 demultiplex DS1

Dependencies The instrument must be set up to receive or add a DS3 tributary signal for this command to apply.

Errors and Events None

Examples
```
INPUT3:TELECOM:RATE DS3
```

Related Commands SENSe:DATA:TELecom:SOURce
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel

INPUT3:TELecom:RATE?

Add/Drop/Test Option Only

This query returns the current setting of the DS3 or DS3_DEMUX tributary input rate.

Syntax
```
INPUT3:TELecom:RATE?
```

Response

<trib2 input rate> (discrete)	description
DS3	44.736 Mb/s (default)
DS3_DEMUX	44.736 Mb/s DS3 demultiplex DS1

Dependencies None

Errors and Events None

Examples Query: `INPUT3:TELECOM:RATE?`

Response: `DS3`

Related Commands INPUT3:TELecom:RATE
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel

INPUT3:TELecom:STATus?

Add/Drop/Test Option Only

This query returns the status of the received or added DS3 tributary signal.

Syntax `INPUT3:TELecom:STATus?`

Response

<trib2 input status> (discrete)	description
NORMal	Signal is of acceptable quality
LOSignal	Loss of Signal (no signal connected)

Dependencies SENSe:DATA:TELecom:SOURce must be set to INPUT3 for this query to apply.

Errors and Events None

Examples Query: `INPUT3:TELECOM:STATUS?`

Response: `NORMAL`

Related Commands SENSe:DATA:TELecom:SOURce

SENSe:DATA:TELecom Subsystem

This section describes the commands and queries that set up the structure of the signal to be received. Figure 2–24 shows the hierarchy tree for this subsystem.

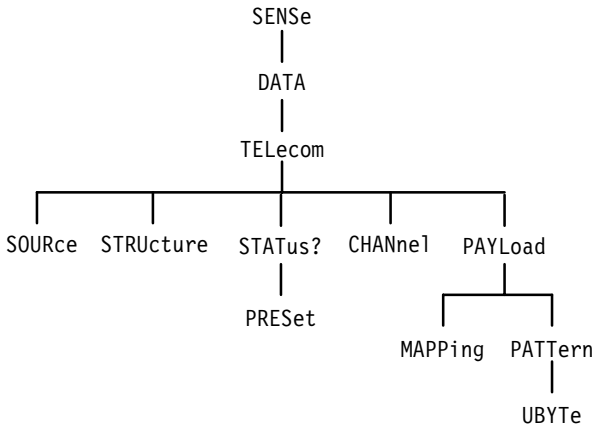


Figure 2–24: SENSe:DATA:TELecom Subsystem

SENSe:DATA:TELecom:SOURce

This command selects the input signal source.

Syntax `SENSe:DATA:TELecom:SOURce <source>`

Parameters

<source> (discrete)	description
INPUT1	SONET rates (default)
INPUT2	DS1 rate (Add/Drop/Test Option Only)
INPUT3	DS3 rate (Add/Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:SOURCE INPUT1`

Related Commands INPUT1:TELecom:RATE

SENSe:DATA:TELecom:SOURce?

This query returns the current setting of the input signal source.

Syntax `SENSe:DATA:TELecom:SOURce?`

Response

<source> (discrete)	description
INPUT1	SONET rates (default)
INPUT2	DS1 rate (Add/Drop/Test Option Only)
INPUT3	DS3 rate (Add/Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples Query: `SENSe:DATA:TELECOM:SOURce?`

Response: `INPUT1`

Related Commands SENSe:DATA:TELecom:SOURce

SENSe:DATA:TELecom:STRUcture

This command selects the input signal structure.

Syntax `SENSe:DATA:TELecom:STRUcture <input structure>`

Parameters

<input structure> (discrete)	description
STS1	STS-1 structure (default)
STS3C	STS-3c structure

Dependencies Selection of STS3C requires OUTPUT1:TELecom:RATE to be set to STS3 or STS12.

Errors and Events 221, “Settings conflict; Argument not valid in current instrument state”

Examples `SENSE:DATA:TELECOM:STRUCTURE STS1`

Related Commands OUTPUT1:TELecom:RATE

SENSe:DATA:TELecom:STRUcture?

This query returns the selected input signal structure.

Syntax `SENSe:DATA:TELecom:STRUcture?`

Response

<input structure> (discrete)	description
STS1	STS-1 structure (default)
STS3C	STS-3c structure

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:STRUCTURE?`

Response: `STS1`

Related Commands SENSe:DATA:TELecom:STRUcture

SENSe:DATA:TELecom:STATus?

This query returns the historical or accumulated status of the received signal.

Syntax `SENSe:DATA:TELecom:STATus?`

Response

<decimal value> (NR1-numeric)	bit	definition
1	0	LOS
2	1	LOF

(continued on next page)

<decimal value> (NR1-numeric)	bit	definition
4	2	OOF
8	3	LOP
16	4	Line AIS
32	5	Path AIS
64	6	Error
128	7	Undefined
256	8	K1/K2 change
512	9	Line FERF
1024	10	Path FERF
2048	11	Pointer adjust
4096	12	NDF
8192	13	Pattern lock
16384	14	Not used
32768	15	Not used

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:STATUS?`

Response: `1024`

Related Commands SENSe:DATA:TELecom:STATus:PRESet

SENSe:DATA:TELecom:STATus:PRESet

This command clears the status of the received SONET and tributary signals by setting each status bit to 0. After this command is given, the status information is accumulated until another SENSe:DATA:TELecom:STATus:PRESet command is given. To get the current signal status without any history information, send the SENSe:DATA:TELecom:STATus:PRESet;:SENSe:DATA:TELecom:STATus? chained command.

Syntax `SENSe:DATA:TELecom:STATus:PRESet`

Parameters None

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:STATUS:PRESET`

Related Commands SENSe:DATA:TELecom:STATus?

SENSe:DATA:TELecom:CHANnel

This command selects the active channel to test (an STS structure with its pointers). For example, in SONET, an STS-3 signal may have three STS-1 signals multiplexed into it. This command allows you to choose one of these three to test.

Syntax `SENSe:DATA:TELecom:CHANnel <channel>`

Parameters

<channel> (NR1-numeric)	description
1	STS-1 rate or STS-3 rate with STS-3c structure (default)
1 to 3	STS-3 rate with STS-1 structure
1 to 4	STS-12 rate with STS-3c structure
1 to 12	STS-12 rate with STS-1 structure

Dependencies Selection of a <channel> greater than 1 implies a rate and structure with multiple STSs.

Errors and Events 221, “Settings conflict; Channel is out of range”

Examples

```
SENSE:DATA:TELECOM:CHANNEL 1
```

Related Commands INPUT1:TELecom:RATE
SENSe:DATA:TELecom:STRUcture

SENSe:DATA:TELecom:CHANnel?

This returns the selected active channel under test.

Syntax

```
SENSe:DATA:TELecom:CHANnel?
```

Response

<channel> (NR1-numeric)	description
1	STS-1 rate or STS-3 rate with STS-3c structure (default)
1 to 3	STS-3 rate with STS-1 structure
1 to 4	STS-12 rate with STS-3c structure
1 to 12	STS-12 rate with STS-1 structure

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:CHANNEL?`

Response: 1

Related Commands SENSe:DATA:TELecom:CHANnel

SENSe:DATA:TELecom:PAYLoad:MAPPing

This command selects the payload mapping of the received structure.

Syntax

```
SENSe:DATA:TELecom:PAYLoad:MAPPing <payload_mapping>
```

Parameters

<mapping> (discrete)	description
EQUipped	Expecting equipped payload mapping (default)
UNEQuipped	Expecting unequipped payload mapping
TRIButary	Expecting tributary payload mapping (Add/ Drop/Test Option Only)

Dependencies Select EQUipped or UNEQuipped to use the SENSe:DATA:TELecom: PAYLoad:PATTern command.

Errors and Events 221, “Settings conflict; Not available without tributary option”

Examples

```
SENSE:DATA:TELECOM:PAYLOAD:MAPPing EQUIPPED
```

Related Commands SENSe:DATA:TELecom:PAYLoad:PATTern

SENSe:DATA:TELecom:PAYLoad:MAPPing?

This query returns the current setting of payload mapping for the received signal.

Syntax

```
SENSe:DATA:TELecom:PAYLoad:MAPPing?
```

Response

<mapping> (discrete)	description
EQUipped	Expecting equipped payload mapping (default)
UNEQuipped	Expecting unequipped payload mapping
TRIButary	Expecting tributary payload mapping (Add/ Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:PAYLOAD:MAPPING?`

Response: `EQUIPPED`

Related Commands SENSe:DATA:TELecom:PAYload:MAPPing

SENSe:DATA:TELecom:PAYLoad:PATTern

This command selects the payload pattern that will be used to calculate the payload BER of the incoming data.

Syntax `SENSe:DATA:TELecom:PAYLoad:PATTern <pattern>`

Parameters

<pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 (default)
PRBS9	A pseudo-random binary sequence of length 2^{9}–1
PRBS15	A pseudo-random binary sequence of length 2^{15}–1
PRBS20	A pseudo-random binary sequence of length 2^{20}–1
AZERos	All zeros
AONEs	All ones
UBYTe	A user-defined byte
UNKNown	Disable BER calculations on incoming data

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:PAYLOAD:PATTERN PRBS23`

Related Commands SENSe:DATA:TELecom:CHANnel
SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe

SENSe:DATA:TELecom:PAYLoad:PATTern?

This query returns the selected payload pattern that is being used to calculate the payload BER in incoming data.

Syntax `SENSe:DATA:TELecom:PAYLoad:PATTern?`

Response

<pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 (default)
PRBS9	A pseudo-random binary sequence of length 2^{9}–1
PRBS15	A pseudo-random binary sequence of length 2^{15}–1
PRBS20	A pseudo-random binary sequence of length 2^{20}–1
AZERos	All zeros
AONEs	All ones
UBYTe	A user-defined byte
UNKNown	Disable BER calculations on incoming data

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:PAYLOAD:PATTERN?`

Response: `PRBS23`

Related Commands SENSe:DATA:TELecom:PAYLoad:PATTern

SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe

This command selects the internally generated payload fixed pattern to be detected in the incoming signal.

Syntax `SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe <fixed pattern>`

Parameters

<fixed pattern> (NR1-numeric)[1]	description
A number in the range 0 to 255 (hexadecimal 00 to FF)	The payload pattern to be detected is set to this value (default = 0)

1 **A hexadecimal value is also acceptable.**

Dependencies SENSe:DATA:TELecom:PAYLoad:PATTern must be set to UBYTe for this command to apply.

Errors and Events None

Examples

```
SENSE:DATA:TELECOM:PAYLOAD:PATTERN:UBYTE 01
SENSE:DATA:TELECOM:PAYLOAD:PATTERN:UBYTE #HA5
```

Related Commands SENSe:DATA:TELecom:PAYLoad:PATTern

SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe?

This query returns the selected internally generated payload fixed pattern to be detected in the incoming signal.

Syntax

```
SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe?
```

Response

<fixed pattern> (NR1-numeric)	description
A number in the range 0 to 255	The payload pattern to be detected is set to this value (default = 0)

Dependencies None

Errors and Events None

Examples

Query: `SENSE:DATA:TELECOM:PAYLOAD:PATTERN:UBYTE?`

Response: `123`

Related Commands SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe

SENSe:DATA:TELecom:TEST Subsystem

This section describes each of the commands and queries used to control measurements. Figure 2–25 shows the hierarchy tree for this subsystem.

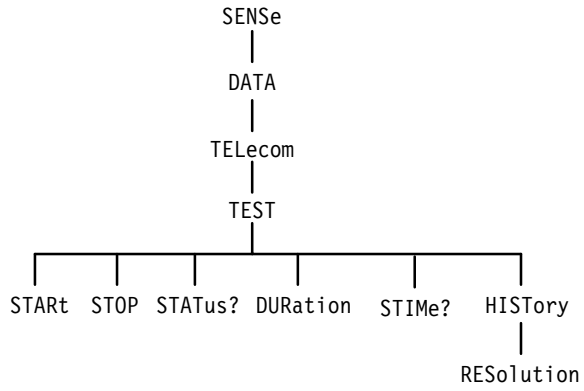


Figure 2–25: SENSe:DATA:TELecom:TEST Subsystem

SENSe:DATA:TELecom:TEST:STARt

This command starts the test measurements.

Syntax `SENSe:DATA:TELecom:TEST:STARt`

Parameters None

Dependencies Any instrument settings can be changed after a test is started, but the measurements are restarted when any change is made to the receiver rate, level, structure, or pattern.

Errors and Events None

Examples `SENSE:DATA:TELECOM:TEST:START`

Related Commands SENSe:DATA:TELecom:TEST:STOP

SENSe:DATA:TELecom:TEST:STOP

This command stops the test measurements.

Syntax `SENSe:DATA:TELecom:TEST:STOP`

Parameters None

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:TEST:STOP`

Related Commands SENSe:DATA:TELecom:TEST:STARt

SENSe:DATA:TELecom:TEST:STATus?

This query returns the state of the measurement process and how long the test has been running.

Syntax `SENSe:DATA:TELecom:TEST:STATus?`

Response

<status> (boolean)	**description**
1	Test is running
0	Test is stopped
<days> (NR1-numeric)	**description**
Any number in the range 0 to 999	Number of days the test has been running
<hours> (NR1-numeric)	**description**
Any number in the range 0 to 23	Number of hours the test has been running
<minutes> (NR1-numeric)	**description**
Any number in the range 0 to 59	Number of minutes the test has been running
<seconds> (NR1-numeric)	**description**
Any number in the range 0 to 59	Number of seconds the test has been running

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TEST:STATUS?`

Response: `1,0,0,13,5`

This response indicates that the test has been running for 13 minutes and 5 seconds and is still running.

Related Commands None

SENSe:DATA:TELecom:TEST:DURation

This command specifies the length of the test. If all four parameters are set to 0, the test will run continuously.

Syntax `SENSe:DATA:TELecom:TEST:DURation <days>,<hours>,<minutes>, <seconds>`

Parameters

<days> (NR1-numeric)	**description**
Any number in the range 0 to 99	Specifies the number of days the test is to be run (default = 0)
<hours> (NR1-numeric)	**description**
Any number in the range 0 to 23	Specifies the number of hours the test is to be run (default = 0)
<minutes> (NR1-numeric)	**description**
Any number in the range 0 to 59	Specifies the number of minutes the test is to be run (default = 0)
<seconds> (NR1-numeric)	**description**
Any number in the range 0 to 59	Specifies the number of seconds the test is to be run (default = 0)

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:TEST:DURATION 1,12,30,0`

This test duration is set to 1 day, 12 hours, 30 minutes, and 0 seconds.

Related Commands None

SENSe:DATA:TELecom:TEST:DURation?

This query returns the value of the test length. If all four parameters are set to 0, the test will run continuously.

Syntax `SENSe:DATA:TELecom:TEST:DURation?`

Response

<days> (NR1-numeric)	description
Any number in the range 0 to 99	Specifies the number of days the test is to be run
<hours> (NR1-numeric)	**description**
Any number in the range 0 to 23	Specifies the number of hours the test is to be run
<minutes> (NR1-numeric)	**description**
Any number in the range 0 to 59	Specifies the number of minutes the test is to be run
<seconds> (NR1-numeric)	**description**
Any number in the range 0 to 59	Specifies the number of seconds the test is to be run

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TEST:DURATION?`

Response: `0,0,15,0`

Related Commands SENSe:DATA:TELecom:TEST:DURation

SENSe:DATA:TELecom:TEST:STIMe?

This query returns the date and time the test was started.

Syntax `SENSe:DATA:TELecom:TEST:STIMe?`

Response

<year> (NR1-numeric)	description
Any number in the range 00 to 99	Specifies the year the test was started; “92” indicates that the test was started in 1992, “01” indicates the year 2001
<month> (NR1-numeric)	**description**
Any number in the range 1 to 12	Specifies the month the test was started; “09” indicates that the test was started in September
<day> (NR1-numeric)	**description**
Any number in the range 1 to 31	Specifies the day of the month the test was started
<hours> (NR1-numeric)	**description**
Any number in the range 0 to 23	Specifies the hour the test was started
<minutes> (NR1-numeric)	**description**
Any number in the range 0 to 59	Specifies the minute the test was started
<seconds> (NR1-numeric)	**description**
Any number in the range 0 to 59	Specifies the seconds the test was started

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TEST:STIME?`

Response: `93,10,25,22,15,00`

This test was started on October 25, 1993 at 10:15 pm.

Related Commands SENSe:DATA:TELecom:TEST:STARt

SENSe:DATA:TELecom:TEST:HISTory:RESolution

This command sets the resolution of the history data.

This command does not affect previously acquired history data. This command assigns the size of data "buckets" prior to starting a new test.

Syntax `SENSe:DATA:TELecom:TEST:HISTory:RESolution <hist res>`

Parameters

<hist res> (discrete)	description
MIN1	1 minute resolution (default)
MIN15	15 minute resolution

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:TEST:HISTORY:RESOLUTION MIN15`

Related Commands SENse:DATA:TELecom:TEST:START
SENse:DATA:TELecom:MEASURE:HISTORY

SENSe:DATA:TELecom:TEST:HISTory:RESolution?

This query returns the resolution of the history data. This returns the size of data "buckets" for the next test.

Syntax `SENSe:DATA:TELecom:TEST:HISTory:RESolution?`

Response

<hist res> (discrete)	description
MIN1	1 minute resolution (default)
MIN15	15 minute resolution

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TEST:HISTORY:RESOLUTION MIN15`

Response: `MIN15`

Related Commands SENse:DATA:TELecom:TEST:START
SENse:DATA:TELecom:MEASURE:HISTORY
SENSe:DATA:TELecom:TEST:HISTory:RESolution

SENSe:DATA:TELecom:OVERhead and POVerhead Subsystem

This section describes each of the commands and queries used to analyze the transport overhead and path overhead. Figure 2–26 shows the hierarchy tree for this subsystem.

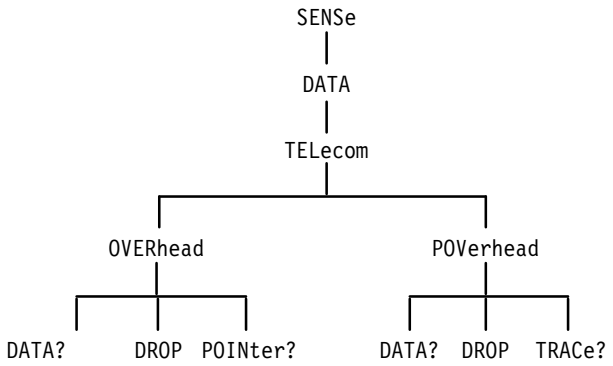


Figure 2–26: SENSe:DATA:TELecom:OVERhead and POVerhead Subsystem

SENSe:DATA:TELecom:OVERhead:DATA?

This query returns the value in the transport overhead byte of the specified channel. See Figures 2–14 and 2–15 on pages 2–38 and 2–39 for the definition of offset values.

Syntax `SENSe:DATA:TELecom:OVERhead:DATA? <channel>,<byte>,<offset>`

Parameters

<channel> (NR1-numeric)	description
1	Rate is STS-1
1 to 3	Rate is STS-3 with STS-1 structure
1 to 4	Rate is STS-12 with STS-3c structure
1 to 12	Rate is STS-12 with STS-1 structure
<byte> (discrete)	**description**
A1, A2, B1, B2, H1, H2, H3, C1, D1, F1, D1, D2, D3, K1, K2, D4, D5, D6, D7, D8, D9, D10, D11, D12, Z1, Z2, E2	Only the bytes listed return a valid response

(continued on next page)

<offset> (NR1-numeric)	description
0	STS-1 structure
0 to 2	STS-3c structure

Response

<value> (NR1-numeric)	description
Any number in the range 0 to 255	The byte is set to this value
–1	No signal received

Dependencies None

Errors and Events 221, “Settings conflict; Channel is out of range”

Examples Query: `SENSE:DATA:TELECOM:OVERHEAD:DATA? 1,C1,O`

Response: `123`

Related Commands INITiate
TRIGger:IMMediate

SENSe:DATA:TELecom:OVERhead:DROP

This command selects the transport overhead bytes to be dropped to an external protocol analyzer.

Syntax `SENSe:DATA:TELecom:OVERhead:DROP <dropped overhead>`

Parameters

<dropped overhead> (discrete)	description
NONE	No overhead bytes dropped (default)
SDCC	Section DCC
LDCC	Line DCC
F1	F1 User Byte

Dependencies You can drop the transport overhead or the path overhead by using the SENSe:DATA:TELecom:OVERhead:DROP and SENSe:DATA:TELecom:POVerhead:DROP commands. The last command sent applies.

Errors and Events None

Examples
```
SENSE:DATA:TELECOM:OVERHEAD:DROP SDCC
```

Related Commands SENse:DATA:TELecom:POVerhead:DROP

SENSe:DATA:TELecom:OVERhead:DROP?

This query returns the current setting of the transport overhead bytes to be dropped to an external protocol analyzer.

Syntax
```
SENSe:DATA:TELecom:OVERhead:DROP?
```

Response

<dropped overhead> (discrete)	description
NONE	No overhead bytes dropped (default)
SDCC	Section DCC
LDCC	Line DCC
F1	F1 User Byte

Dependencies None

Errors and Events None

Examples Query:
```
SENSE:DATA:TELECOM:OVERHEAD:DROP?
```
Response:
```
NONE
```

Related Commands SENSe:DATA:TELecom:OVERhead:DROP

SENSe:DATA:TELecom:OVERhead:POINter?

This query returns the current value of the H1 and H2 overhead bytes of the active channel. If the instrument is receiving a LOS, LOF, or LOP, the last valid pointer value will be returned.

Syntax `SENSe:DATA:TELecom:OVERhead:POINter?`

Response

<pointer value> (NR1-numeric)	description
Any integer in the range 0 to 1023	H1 and H2 are set to this value

Dependencies You must have a test running for a valid pointer value to be returned (use the SENSe:DATA:TELecom:TEST:STARt command to start a test).

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:OVERHEAD:POINTER?`

Response: `123`

Related Commands SENSe:DATA:TELecom:OVERhead:DATA?
SENSe:DATA:TELecom:TEST:STARt

SENSe:DATA:TELecom:POVerhead:DATA?

This query returns the value in the specified path overhead byte.

Syntax `SENSe:DATA:TELecom:POVerhead:DATA? <byte>`

Parameters

<byte> (discrete)	description
J1, B3, C2, G1, F2, H4, Z3, Z4, Z5	Only the bytes listed are available for selection

Response

<value> (NR1-numeric)	description
Any number in the range 0 to 255	The byte is set to this value (the value for J1 is the ASCII representation of the string value)

Dependencies Use the SENSe:DATA:TELecom:CHANNel command to specify which path trace to query.

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:POVERHEAD:DATA? C2`

Response: `123`

Related Commands INITiate
TRIGger:IMMediate

SENSe:DATA:TELecom:POVerhead:DROP

This command selects the path overhead channels to be dropped to an external protocol analyzer.

Syntax `SENSe:DATA:TELecom:POVerhead:DROP <dropped overhead>`

Parameters

<dropped overhead> (discrete)	description
NONE	Nothing is dropped (default)
F2	F2 User Byte

Dependencies You can drop the transport overhead or the path overhead by using the SENSe:DATA:TELecom:OVERhead:DROP and SENSe:DATA:TELecom:POVerhead:DROP commands. The last command sent applies.

Errors and Events None

Examples

```
SENSE:DATA:TELECOM:POVERHEAD:DROP F2
```

Related Commands SENSe:DATA:TELecom:OVERhead:DROP

SENSe:DATA:TELecom:POVerhead:DROP?

This query returns the current state of the dropped path overhead.

Syntax

```
SENSe:DATA:TELecom:POVerhead:DROP?
```

Response

<dropped overhead> (discrete)	description
NONE	Nothing is dropped (default)
F2	F2 User Byte

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:POVERHEAD:DROP?`

Response: `NONE`

Related Commands SENSe:DATA:TELecom:POVerhead:DROP

SENSe:DATA:TELecom:POVerhead:TRACe?

This query returns the current path trace string that repeats in the J1 byte as a repeating byte sequence. The response is created in the following way: the first character after a null is read as the first byte and is followed by 63 J1 bytes from consecutive frames.

Syntax

```
SENSe:DATA:TELecom:POVerhead:TRACe?
```

Response

<path trace> (string)	description
Length is a maximum of 64 bytes; if length is less than 64 bytes, the buffer is padded with nulls	The J1 byte is set to this value

Dependencies

The SENSe:DATA:TELecom:CHANnel command specifies which path trace to query.

Errors and Events

None

Examples

Query: `SENSE:DATA:TELECOM:POVERHEAD:TRACE?`

Response: `"THIS IS THE FIRST RUN OF TEST ABC"`

Related Commands

SENSe:DATA:TELecom:POVerhead:DATA?
INITiate
TRIGger:IMMediate

SENSe:DATA:TELecom:MEASure Subsystem

This section describes the commands and queries that access error, alarm, failure, and pointer measurements for current and previous tests. Figures 2–27, 2–28, 2–29, 2–30, and 2–31 show the hierarchy trees for this subsystem.

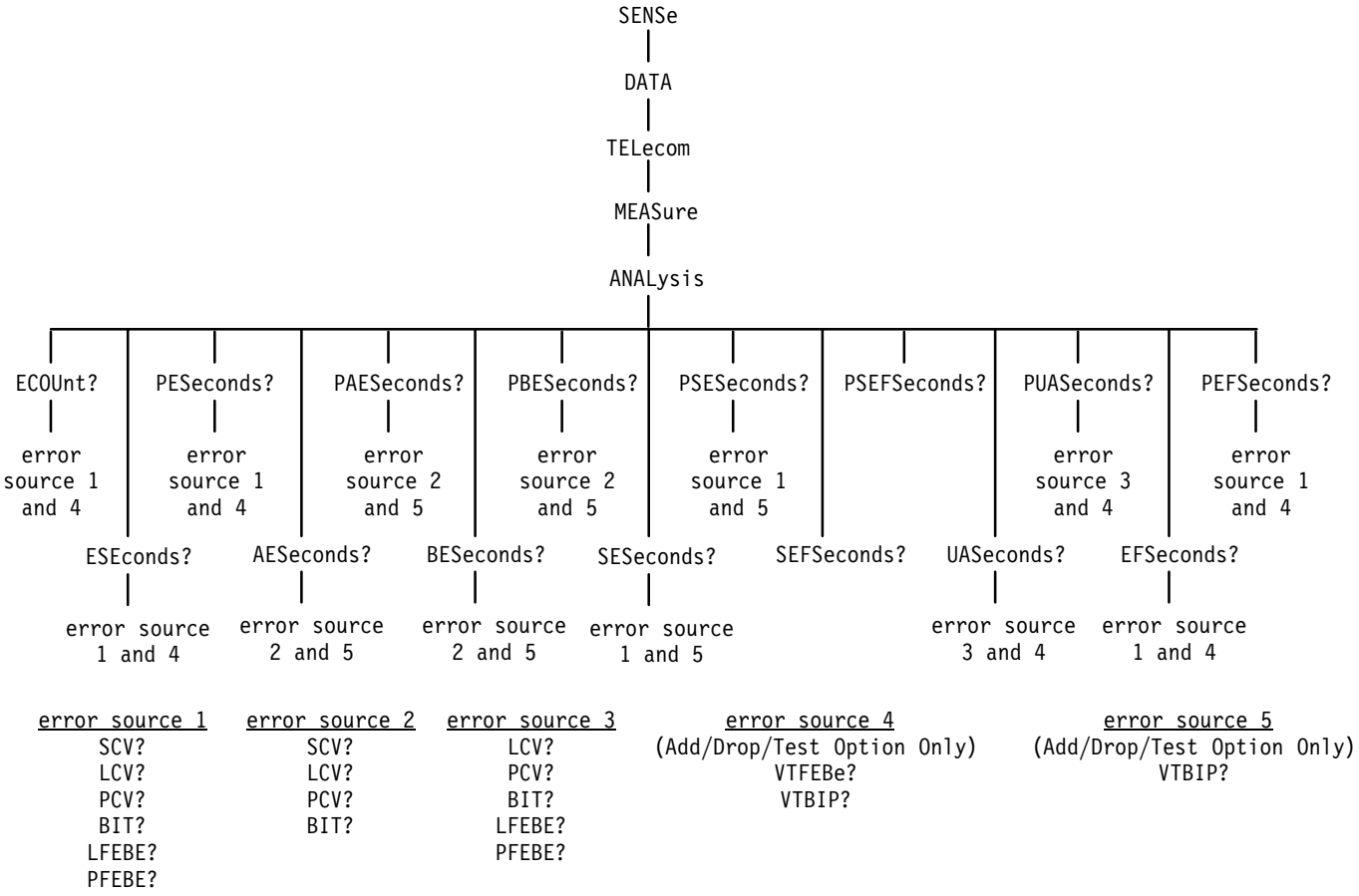


Figure 2–27: SENSe:DATA:TELecom:MEASure:ANALysis Subsystem (SONET)

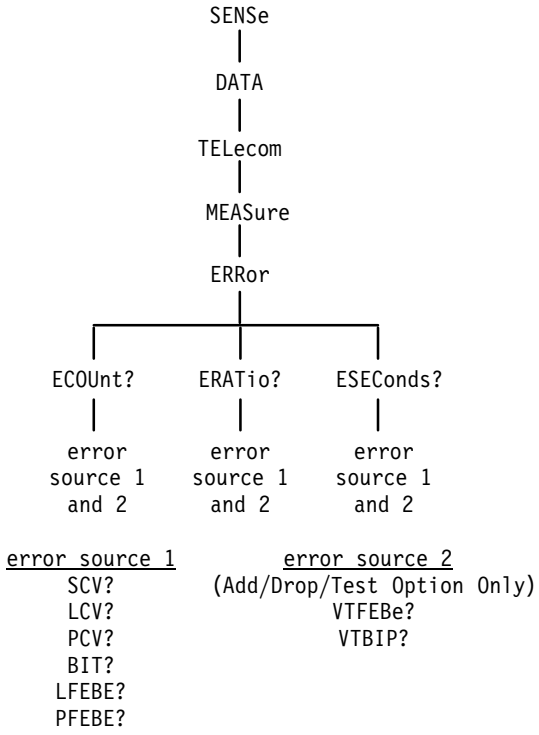


Figure 2–28: SENSe:DATA:TELecom:MEASure:ERRor Subsystem

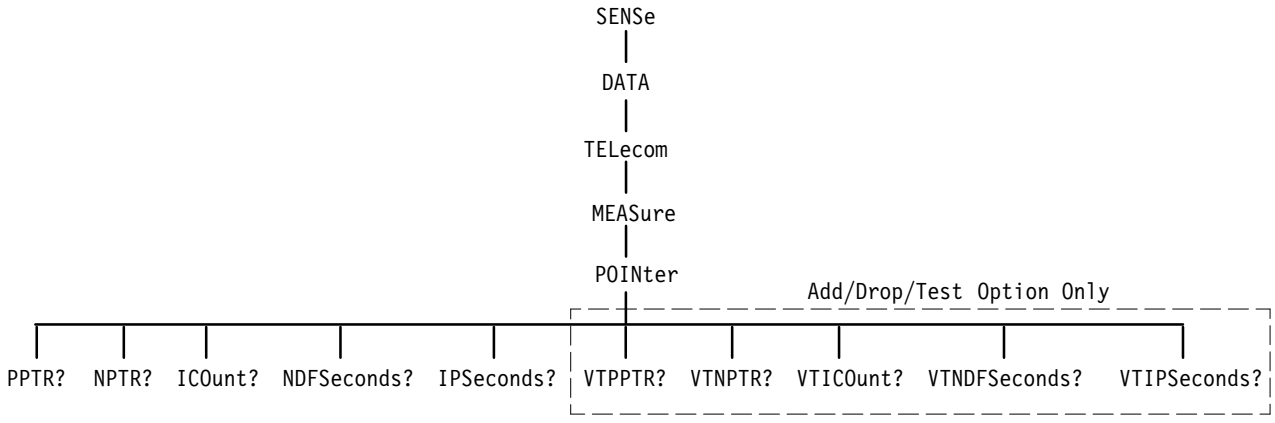


Figure 2–29: SENSe:DATA:TELecom:MEASure:POINter Subsystem

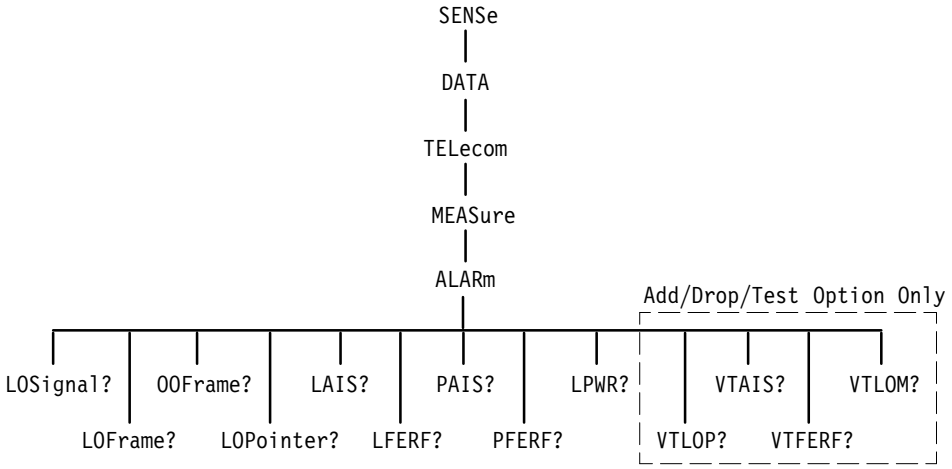


Figure 2–30: SENSe:DATA:TELecom:MEASure:ALARm Subsystem

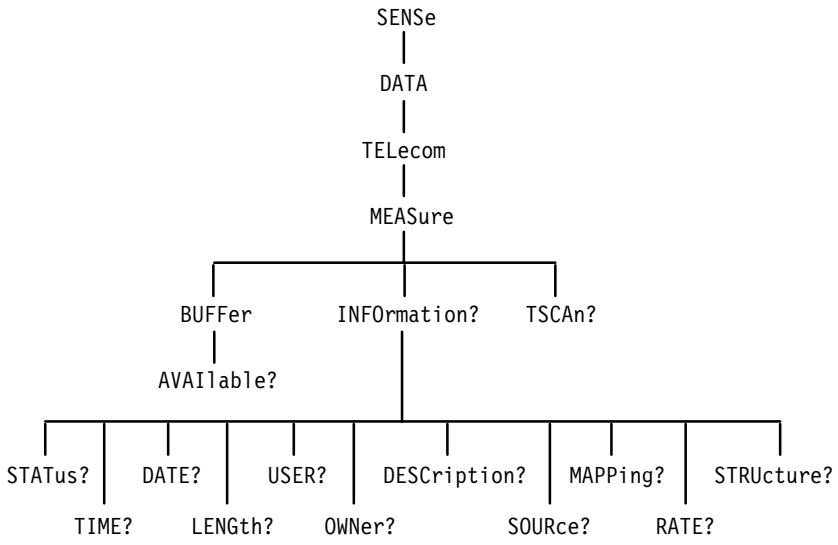


Figure 2–31: SENSe:DATA:TELecom:MEASure:BUFFer and INFOrmation Subsystems

A variety of error, alarm, failure, and pointer measurements are reported through this subsystem. Table 2–7 shows how error, alarm, and failure measurements are calculated. Tables 2–8 and 2–9 show how the analysis measurements are calculated. These calculations are based on T1M1.93 specifications.

Table 2–7: How Error, Alarm, and Pointer Measurements are Calculated

Type of Measurement	Method of Calculation
Error count	Number of bit errors that were errored in the signal
Bit Error Ratio (BER)	Ratio of error count to the total number of received bits
Errored seconds	Number of seconds that had any error counts or failures such as Loss of Signal (LOS) or Severely Errored Frame (SEF)
Pointer measurements	Number of events that occur in the H1 and H2 pointer bytes
New Data Flag Seconds	Number of one-second intervals that contain new data flags
Illegal Pointer Seconds	Number of one-second intervals that contain illegal pointers
Positive Pointer Justifications	Number of times the pointer value is incremented
Negative Pointer Justifications	Number of times the pointer value is decremented
Alarms	Number of one-second intervals that contained a specific alarm such as Loss of Signal (LOS), Loss of Pointer (LOP), and Path Alarm Indication Signal (PAIS)

Table 2–8: How Analysis Measurements are Calculated

Type of Measurement	Method of Calculation
Error count	Number of bit errors not occurring during periods of unavailability (see Unavailable seconds)
Errored seconds	Total number of type A, type B, and severely errored seconds not occurring during a period of unavailability (see Unavailable seconds)
Type A errored seconds	Number of seconds that had exactly one error count
Type B errored seconds	Number of seconds that had more than one error count and less than N errors (see Table 2–9)
Severely errored seconds	Number of seconds with more than N errors (see Table 2–9)
Severely errored framing seconds	Number of seconds where the incoming signal could not be framed (applies only to the Section layer)
Unavailable seconds	Number of seconds that the signal had too many errors to be available for use; unavailability starts at the onset of ten contiguous severely errored seconds
Error free seconds	Number of seconds that contained zero errors

Table 2–9: Value of N for Analysis Measurements

Rate	N (Section B1 Errors)	N (Line B2 Errors)
STS-1	2500	2500
STS-3	2500	2500
STS-12	8800	10000

Some of the queries in this section have their information presented in a way that is different from queries in the rest of the manual. The syntax and examples are in table format. Figure 2–32 shows you how to read the Syntax Tables in this section. Follow the step numbers to create any query. Table 2–10 explains the acronyms used in the SENSe:DATA:TELecom:MEASure Syntax Tables.

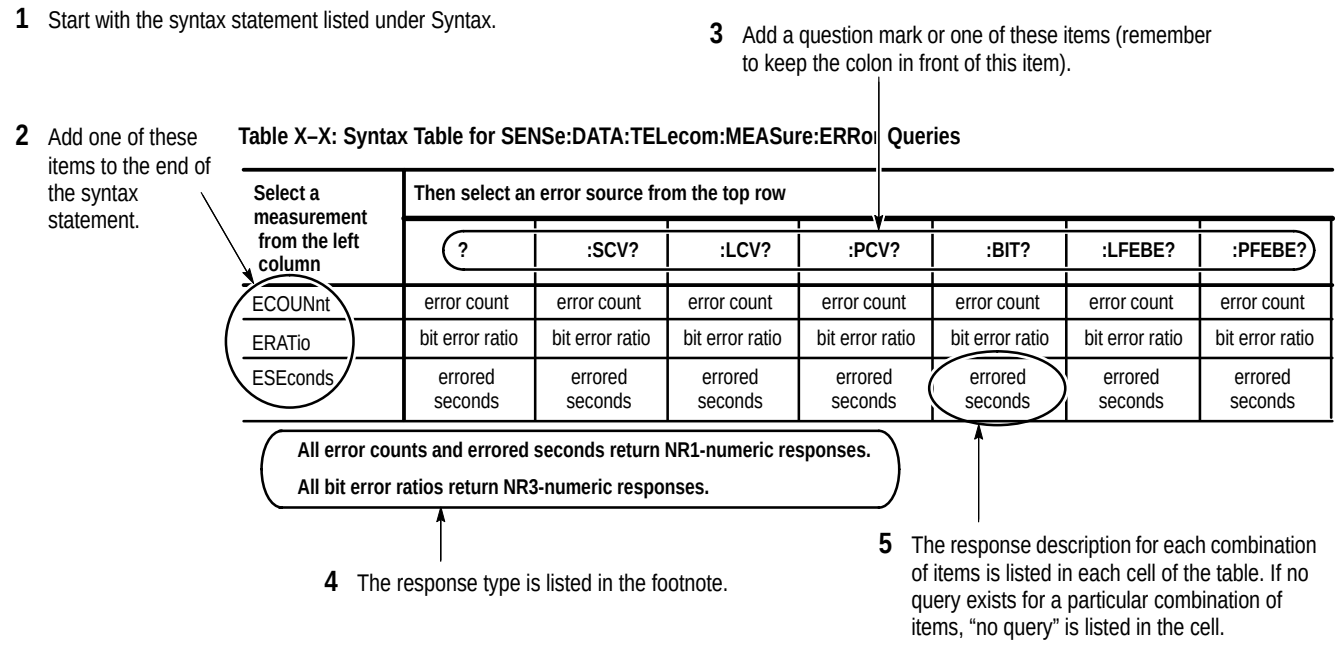


Figure 2–32: How to Read the Syntax Tables in the SENSe:DATA:TELecom:MEASure Subsystem Section

Table 2–10: Acronyms Used in the SENSe:DATA:TELecom:MEASure Queries

Acronym	Meaning
SCV	Section/RS Code Violation
LCV	Line/MS Code Violation
PCV	Path Code Violation
BIT error (not an acronym)	Pattern bit error
LFEBE	Line/MS Far End Block Error
PFEBE	Path Far End Block Error
VTBIP	VT BIP (Add/Drop/Test Option Only)
VTFEBE	VT FEBE (Add/Drop/Test Option Only)

Figure 2–33 shows you how to read the Example Tables in this section.

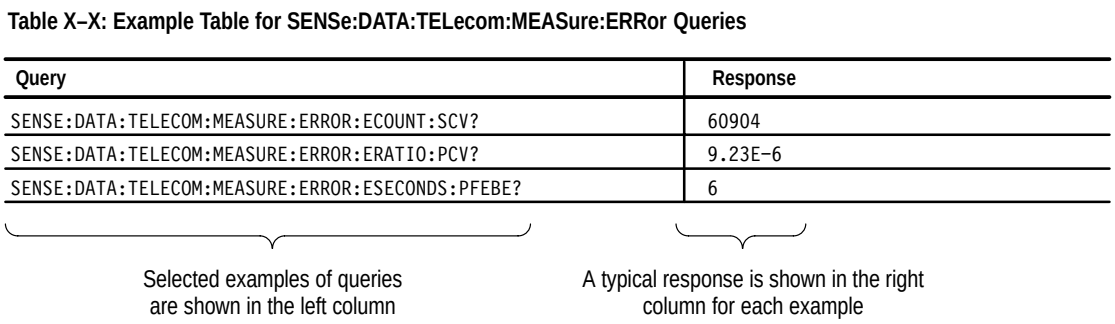


Figure 2–33: How to Read the Example Tables in the SENSe:DATA:TELecom:MEASure Subsystem Section

SENSe:DATA:TELecom:MEASure:ERRor Queries

These queries return error measurements. When you use the high-level queries (for example, SENSe:DATA:TELecom:MEASure:ERRor? or SENSe:DATA:TELecom:MEASure:ERRor:ECOUnt?), it is helpful to turn the headers on (SYSTem:HEADers ON) so you can identify each response value in the response string.

Syntax

```
SENSe:DATA:TELecom:MEASure:ERRor?
```

```
SENSe:DATA:TELecom:MEASure:ERRor:[measurement]:[error source]
```

(see Tables 2–11 and 2–12 to complete the query)

Table 2–11: Syntax Table for SENSe:DATA:TELecom:MEASure:ERRor Queries

Select a measurement from the left column	Then select an error source from the top row						
	?	:SCV?	:LCV?	:PCV?	:BIT?	:LFEBE?	:PFEBE?
ECOUnt	all error counts	error count	error count	error count	error count	error count	error count
ERATio	all bit error ratios	bit error ratio	bit error ratio	bit error ratio	bit error ratio	bit error ratio	bit error ratio
ESEConds	all errored seconds	errored seconds	errored seconds	errored seconds	errored seconds	errored seconds	errored seconds

All error counts and errored seconds return NR1-numeric responses.

All bit error ratios return NR3-numeric responses.

Table 2–12: Syntax Table for SENSe:DATA:TELecom:MEASure:ERRor Queries (Add/Drop/Test Option Only)

Select a measurement from the left column	Then select an error source from the top row	
	:VTFEBe?	:VTBIP?
ECOUnt:	error count	error count
ERATio	bit error ratio	bit error ratio
ESEConds	errored seconds	errored seconds

All error counts and errored seconds return NR1-numeric responses.

All bit error ratios return NR3-numeric responses.

Response See Tables 2–11 and 2–12.

Dependencies These measurement queries can be sent at any time. But, if a test is currently running, the responses to the queries might not represent the final error measurements. After a test has been stopped or the test duration has expired, you can send these measurement queries again to get the final error measurements.

Errors and Events None

Examples See Table 2–13.

Table 2–13: Example Table for SENSe:DATA:TELecom:MEASure:ERROr Queries

Query	Response
SENSE:DATA:TELECOM:MEASURE:ERROR:ECOUNT:SCV?	60904
SENSE:DATA:TELECOM:MEASURE:ERROR:ERATIO:PCV?	9.23E-6
SENSE:DATA:TELECOM:MEASURE:ERROR:ESECONDS:PFEBE?	6

Related Commands SENSe:DATA:TELecom:TEST:STARt
SENSe:DATA:TELecom:TEST:STOP

SENSe:DATA:TELecom:MEASure:ANALysis Queries

These queries return an analysis of section, line, path, and payload errors. When you use the high-level queries (for example, SENSe:DATA:TELecom: MEASure:ANALysis? or SENSe:DATA:TELecom:MEASure: ANALysis:ECOUnt?), it is helpful to turn the headers on (SYSTem:HEADers ON) so you can identify each response value in the response string.

Syntax `SENSe:DATA:TELecom:MEASure:ANALysis?`

`SENSe:DATA:TELecom:MEASure:ANALysis:[measurement]:[error source]` (see Tables 2–14 and 2–15 to complete the query)

Table 2–14: Syntax Table for SENSe:DATA:TELecom:MEASure:ANALysis Queries

Select a measurement from the left column	Then select an error source from the top row						
	?	:SCV?	:LCV?	:PCV?	:BIT?	:LFEBE?	:PFEBE?
ECOUnt	all error counts	error count	error count	error count	error count	error count	error count
ESEConds	all errored seconds	errored seconds	errored seconds	errored seconds	errored seconds	errored seconds	errored seconds
PESeconds	all percent errored seconds	percent errored seconds	percent errored seconds	percent errored seconds	percent errored seconds	percent errored seconds	percent errored seconds
AESeconds	all type A errored seconds	type A errored seconds	type A errored seconds	type A errored seconds	type A errored seconds	no query	no query
PAESeconds	all percent type A errored seconds	percent type A errored seconds	percent type A errored seconds	percent type A errored seconds	percent type A errored seconds	no query	no query

Table 2–14: Syntax Table for SENSe:DATA:TELecom:MEASure:ANALysis Queries (Cont.)

Select a measurement from the left column	Then select an error source from the top row						
	?	:SCV?	:LCV?	:PCV?	:BIT?	:LFEBE?	:PFEBE?
BESeconds	all type B errored seconds	type B errored seconds	type B errored seconds	type B errored seconds	type B errored seconds	no query	no query
PBESeconds	all percent type B errored seconds	percent type B errored seconds	percent type B errored seconds	percent type B errored seconds	percent type B errored seconds	no query	no query
SESeconds	all severely errored seconds	severely errored seconds	severely errored seconds	severely errored seconds	severely errored seconds	severely errored seconds	severely errored seconds
PSESeconds	all percent severely errored seconds	percent severely errored seconds	percent severely errored seconds	percent severely errored seconds	percent severely errored seconds	percent severely errored seconds	percent severely errored seconds
SEFSeconds	severely errored framing seconds	no query	no query	no query	no query	no query	no query
PSEFseconds	percent severely errored framing seconds	no query	no query	no query	no query	no query	no query
UASeconds	all unavailable seconds	no query	unavailable seconds	unavailable seconds	unavailable seconds	unavailable seconds	unavailable seconds
PUASeconds	all percent unavailable seconds	no query	percent unavailable seconds	percent unavailable seconds	percent unavailable seconds	percent unavailable seconds	percent unavailable seconds
EFSeconds	all error free seconds	error free seconds	error free seconds	error free seconds	error free seconds	error free seconds	error free seconds
PEFSeconds	all percent error free seconds	percent error free seconds	percent error free seconds	percent error free seconds	percent error free seconds	percent error free seconds	percent error free seconds

All percent measurements return NR3-numeric responses.

All other measurements return NR1-numeric responses.

Table 2–15: Syntax Table for SENSe:DATA:TELecom:MEASure:ANALysis Queries (SONET and Add/Drop/Test Option Only)

Select a measurement from the left column	Then select an error source from the top row	
	:VTFEBe?	:VTBIP?
ECOUnt	all error counts	all error counts
ESEConds	errored seconds	errored seconds
PESeconds	percent errored seconds	percent errored seconds

Table 2–15: Syntax Table for SENSe:DATA:TELecom:MEASure:ANALysis Queries (SONET and Add/Drop/Test Option Only) (Cont.)

Select a measurement from the left column	Then select an error source from the top row	
	:VTFEBe?	:VTBIP?
AESeconds	no query	type A errored seconds
PAESeconds	no query	percent type A errored seconds
BESeconds	no query	type B errored seconds
PBESeconds	no query	percent type B errored seconds
SESeconds	no query	severely errored seconds
PSESeconds	no query	percent severely errored seconds
UASeconds	unavailable seconds	unavailable seconds
PUASeconds	percent unavailable seconds	percent unavailable seconds
EFSeconds	error free seconds	error free seconds
PEFSeconds	percent error free seconds	percent error free seconds

All bit error ratios and percent measurements return NR3-numeric responses.

All other measurements return NR1-numeric responses.

Response See Tables 2–14 and 2–15.

Dependencies These measurement queries can be sent at any time. However, if a test is currently running, the responses to the queries might not represent the final error measurements. After a test has been stopped or the test duration has expired, you can send these measurement queries again to get the final error measurements.

Errors and Events None

Examples See Table 2–16.

Table 2–16: Example Table for SENSe:DATA:TELecom:MEASure:ANALysis Queries (SONET)

Query	Response
SENSE:DATA:TELECOM:MEASURE:ANALYSIS:ECOUNT:LCV?	76824
SENSE:DATA:TELECOM:MEASURE:ANALYSIS:ESECONDS:BIT?	26

Table 2–16: Example Table for SENSe:DATA:TELecom:MEASure:ANALysis Queries (SONET) (Cont.)

Query	Response
SENSE:DATA:TELECOM:MEASURE:ANALYSIS:PUASECONDS:PFEBE?	6.5E-3
SENSE:DATA:TELECOM:MEASURE:ANALYSIS:SESECONDS:PCV?	0

Related Commands SENSe:DATA:TELecom:TEST:STARt
SENSe:DATA:TELecom:TEST:STOP

SENSe:DATA:TELecom:MEASure:ALARm Queries

These queries return alarm measurements. When you use the SENSe:DATA:TELecom:MEASure:ALARm? query, it is helpful to turn the headers on (SYSTem:HEADers ON) so you can identify each response value in the response string.

Syntax All valid queries are listed in the Syntax column of Tables 2–17 and 2–18.

Table 2–17: Syntax Table for SENSe:DATA:TELecom:MEASure:ALARm Queries

Syntax	Response
SENSe:DATA:TELecom:MEASure:ALARm?	All alarm measurements
SENSe:DATA:TELecom:MEASure:ALARm:LOSignal?	Number of seconds of Loss of Signal
SENSe:DATA:TELecom:MEASure:ALARm:LOFrame?	Number of seconds of Loss of Frame
SENSe:DATA:TELecom:MEASure:ALARm:OOFrame?	Number of seconds of Out of Frame
SENSe:DATA:TELecom:MEASure:ALARm:LOPointer?	Number of seconds of Loss of Pointer
SENSe:DATA:TELecom:MEASure:ALARm:LAIS?	Number of seconds of Line AIS
SENSe:DATA:TELecom:MEASure:ALARm:LFERf?	Number of seconds of Line FERF
SENSe:DATA:TELecom:MEASure:ALARm:PFERf?	Number of seconds of Path FERF
SENSe:DATA:TELecom:MEASure:ALARm:PAIS?	Number of seconds of Path AIS
SENSe:DATA:TELecom:MEASure:ALARm:LPWR?	Number of seconds of instrument power loss during a test

All responses are in NR3-numeric format.

Table 2–18: Syntax Table for SENSe:DATA:TELecom:MEASure:ALARm Queries (Add/Drop/Test Option Only)

Syntax	Response
SENSe:DATA:TELecom:MEASure:ALARm:VTLOP?	Number of seconds of tributary Loss of Pointer
SENSe:DATA:TELecom:MEASure:ALARm:VTAIS?	Number of seconds of tributary AIS
SENSe:DATA:TELecom:MEASure:ALARm:VTFERF?	Number of seconds of tributary FERF
SENSe:DATA:TELecom:MEASure:ALARm:VTLOM?	Number of seconds of tributary Loss of Multiframe

All responses are in NR3-numeric format.

Response See the Response column of Tables 2–17 and 2–18.

Dependencies These measurement queries can be sent at any time. However, if a test is currently running, the responses to the queries might not represent the final error measurements. After a test has been stopped or the test duration has expired, you can send these measurement queries again to get the final error measurements.

Errors and Events None

Examples See Table 2–19.

Table 2–19: Example Table for SENSe:DATA:TELecom:MEASure:ALARm Queries

Query	Response
SENSE:DATA:TELECOM:MEASURE:ALARM:LOPOINTER?	20
SENSE:DATA:TELECOM:MEASURE:ALARM:OOFRAME?	13

Related Commands SENSe:DATA:TELecom:TEST:STARt
SENSe:DATA:TELecom:TEST:STOP

SENSe:DATA:TELecom:MEASure:POINter Queries

These queries return pointer-related measurements. When you use the SENSe:DATA:TELecom:MEASure:POINter? query, it is helpful to turn the headers on (SYSTem:HEADers ON) so you can identify each response value in the response string.

Syntax All valid queries are listed in the Syntax column of Tables 2–20 and 2–21.

Table 2–20: Syntax Table for SENSe:DATA:TELecom:MEASure:POINter Queries

Syntax	Response
SENSe:DATA:TELecom:MEASure:POINter?	All pointer measurements
SENSe:DATA:TELecom:MEASure:POINter:NDFSeconds?	Number of seconds in which one or more NDFs (new data flags) occurred
SENSe:DATA:TELecom:MEASure:POINter:IPSeconds?	Number of seconds in which one or more illegal pointer adjustments occurred
SENSe:DATA:TELecom:MEASure:POINter:PPTR?	Number of positive pointer justifications
SENSe:DATA:TELecom:MEASure:POINter:NPTR?	Number of negative pointer justifications
SENSe:DATA:TELecom:MEASure:POINter:ICOunt?	Number of invalid pointers

All responses are in NR1-numeric format.

Table 2–21: Syntax Table for SENSe:DATA:TELecom:MEASure:POINter Queries (Add/Drop/Test Option Only)

Syntax	Response
SENSe:DATA:TELecom:MEASure:POINter:VTPPTR?	Number of tributary positive pointer justifications
SENSe:DATA:TELecom:MEASure:POINter:VTNPTR?	Number of tributary negative pointer justifications
SENSe:DATA:TELecom:MEASure:POINter:VTICOunt?	Number of tributary invalid pointers
SENSe:DATA:TELecom:MEASure:POINter:VTNDFSeconds?	Number of seconds in which one or more tributary NDFs (new data flags) occurred
SENSe:DATA:TELecom:MEASure:POINter:VTIPSeconds?	Number of seconds in which one or more illegal tributary pointer adjustments occurred

All responses are in NR1-numeric format.

Response See the Response column of Tables 2–20 and 2–21.

Dependencies These measurement queries can be sent at any time. However, if a test is currently running, the responses to the queries might not represent the final error measurements. After a test has been stopped or the test duration has expired, you can send these measurement queries again to get the final error measurements.

Errors and Events None

Examples See Table 2–22.

Table 2–22: Example Table for SENSe:DATA:TELecom:MEASure:POINter Queries

Query	Response
SENSE:DATA:TELECOM:MEASURE:POINTER:PPTR?	12
SENSE:DATA:TELECOM:MEASURE:POINTER:ICOUNT?	0

Related Commands SENSe:DATA:TELecom:TEST:STARt
SENSe:DATA:TELecom:TEST:STOP

SENSe:DATA:TELecom:MEASure:BUFFer

This command sets the buffer that is read with the measurement queries. The following buffers are available for use: buffer number 1 contains results from the most recent test, and buffer number 2 contains results from the previous test. The current test results might overflow into the previous test results buffer (buffer number 2). In that case, only buffer number 1 is available for use. Use the SENSe:DATA:TELecom:MEASure:BUFFer:AVAIlable? query to determine the oldest available buffer.

Buffer number –1 contains results that have been accessed from disk. After you give the MMEMory:LOAD:RESUlts command, the buffer number is set to –1.

Syntax `SENSe:DATA:TELecom:MEASure:BUFFer <results buffer>`

Parameters

<results buffer> (NR1-numeric)	description
1 or 2	Buffer number read with the measurement queries (default = 1)
–1	Information from the MMEMory:LOAD: RESUlts command is stored in this buffer

Dependencies None

Errors and Events 200, “Execution error; Temporary buffer is empty”

Examples

```
SENSE:DATA:TELECOM:MEASURE:BUFFER 2
```

Related Commands

SENSe:DATA:TELecom:MEASure:INFOrmation?
SENSe:DATA:TELecom:MEASure:BUFFer:AVAIlable
MMEMory:LOAD:RESUlts

SENSe:DATA:TELecom:MEASure:BUFFer?

This query returns the buffer number that is read with the measurement queries. If the buffer number is set to 1, the measurement queries will return measurement information for the most recent test. If the buffer number is set to 2, the measurement queries will return measurement information for the previous test. The recent test results might overflow into the previous test results buffer (buffer number 2). In that case, only buffer number 1 is available for use. Use the SENSe:DATA:TELecom:MEASure:BUFFer:AVAIlable? query to determine the oldest available buffer.

Buffer number –1 contains results that have been accessed from disk. After you give the MMEMory:LOAD:RESUlts command, the buffer number is set to –1.

Syntax

```
SENSe:DATA:TELecom:MEASure:BUFFer?
```

Response

<results buffer> (NR1-numeric)	description
1 or 2	Buffer number read with the measurement queries (default = 1)
–1	Information from the MMEMory:LOAD: RESUlts command is stored in this buffer

Dependencies

None

Errors and Events

None

Examples

Query: `SENSE:DATA:TELECOM:MEASURE:BUFFER?`

Response: 1

Related Commands

SENSe:DATA:TELecom:MEASure:BUFFer

SENSe:DATA:TELecom:MEASure:BUFFer:AVAIlable?

This query returns the oldest buffer accessible with the measurement and history queries. The value returned by this query is the maximum value you can use in the SENSe:DATA:TELecom:MEASure:BUFFer command.

Buffer number 1 contains results from the most recent test. Buffer number 2 contains results from the previous test. The current results might overflow into the previous test results buffer (buffer number 2). In that case, only buffer number 1 is available for use.

While a test is running, the response to this query is always 1 because only current test results can be displayed at that time.

Syntax

```
SENSe:DATA:TELecom:MEASure:BUFFer:AVAIlable?
```

Response

<oldest buffer> (NR1-numeric)	description
1 or 2	Oldest buffer number read with the measurement queries (default = 1)

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:MEASURE:BUFFER?`

Response: 2

Related Commands SENSe:DATA:TELecom:MEASure:BUFFer

SENSe:DATA:TELecom:MEASure:INFOrmation Queries

This query returns information on the buffer accessed with the measurement and history queries.

Syntax All valid queries are listed in the Syntax column of Table 2–23.

Table 2–23: Syntax Table for SENSe:DATA:TELecom:MEASure:INFOrmation Queries

Syntax	Response
SENSe:DATA:TELecom:MEASure:INFOrmation?	[All measurement information]
SENSe:DATA:TELecom:MEASure:INFOrmation:STATus?	EMPTY, RECORDING, COMPLETE
SENSe:DATA:TELecom:MEASure:INFOrmation:TIME?	hour,minute [the time the test starts]
SENSe:DATA:TELecom:MEASure:INFOrmation:DATE?	year,month,day [the date the test starts]
SENSe:DATA:TELecom:MEASure:INFOrmation:LENGth?	[Length of the test information in “buckets”]
SENSe:DATA:TELecom:MEASure:INFOrmation:INSTrument?	[Instrument identity]
SENSe:DATA:TELecom:MEASure:INFOrmation:OWNer?	[Same information as the SYSTem:OWNer? query]
SENSe:DATA:TELecom:MEASure:INFOrmation:USER?	[Same information as the SYSTem:USER? query]
SENSe:DATA:TELecom:MEASure:INFOrmation:OPTIons?	[Instrument options]
SENSe:DATA:TELecom:MEASure:INFOrmation:DESCription?	[Description of the test]
SENSe:DATA:TELecom:MEASure:INFOrmation:SOURce?	INPUT1, INPUT2, INPUT3 [SONET or tributary signal]
SENSe:DATA:TELecom:MEASure:INFOrmation:MAPPing?	EQUIpped, UNEQuipped, VTASYNC, DS3, DS3_DEMUX
SENSe:DATA:TELecom:MEASure:INFOrmation:RATE?	STS1, STS3, STS12 DS1, DS3, DS3_DEMUX (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:MEASure:INFOrmation:STRUcture?	STS1, STS3CAU3,
SENSe:DATA:TELecom:MEASure:INFOrmation:FRAMing?	[Tributary framing]
SENSe:DATA:TELecom:MEASure:INFOrmation:RESolution?	MIN1 , MIN15, SEC1 [Resolution of acquired data]

The status, time, date, and length responses are in NR1-numeric format.

All other responses are in string format.

If structure, mapping, and framing do not apply to the received signal, the response is NONE.

Response See the Response column of Table 2–23.

Dependencies Information is valid only when a test is completed. Set SENSe:DATA:TELecom:MEASure:BUFFer to the buffer for which you want information.

Errors and Events 200, “Execution error; Test is still running”

Examples See Table 2–24.

Table 2–24: Example Table for SENSe:DATA:TELecom:MEASure:INFOrmation Queries

Query	Response
SENSE:DATA:TELECOM:MEASURE:INFORMATION:TIME?	14,22,0
SENSE:DATA:TELECOM:MEASURE:INFORMATION:DESCRIPTION?	"PORTLAND TO SEATTLE NETWORK TEST"
SENSE:DATA:TELECOM:MEASURE:INFORMATION:MAPPING?	EQUIPPED

Related Commands SENSe:DATA:TELecom:MEASure:BUFFer

SENSe:DATA:TELecom:MEASure:TSCAn?

This query returns Trouble Scan Information to help you determine the severity of errors during a test. The information in the response is for human interpretation only and is highly variable depending upon the signal being tested and whether or not the test is complete.

Syntax SENSe:DATA:TELecom:MEASure:TSCAn?

Response

<tscan information> (string)	description
A string, maximum length of 256 bytes	Information about a test currently running or complete

Dependencies None

Examples Query: SENSE:DATA:TELECOM:MEASURE:TSCAN?

Response: "NO ALARMS, BER: 1.2E-8"

Related Commands None

SENSe:DATA:TELecom:MEASure:STESts Subsystem

This section describes each of the commands and queries that allow you to apply predefined criteria to test results and determine if the tests passed or failed. Figure 2–34 shows the hierarchy tree for this subsystem.

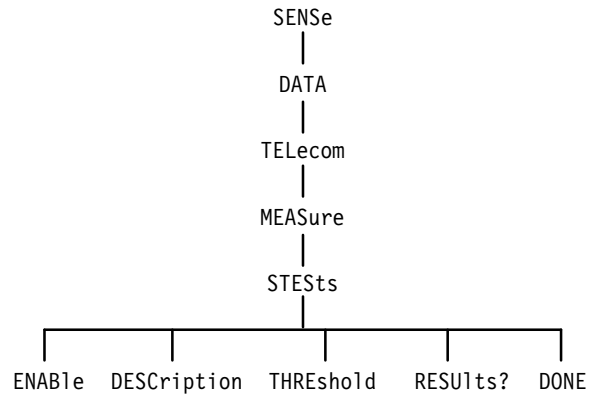


Figure 2–34: SENSe:DATA:TELecom:MEASure:STESts Subsystem

SENSe:DATA:TELecom:MEASure:STESts:ENABle

This command enables the evaluation of pass/fail tests. The enable will revert to OFF (0) after the pass/fail test is evaluated.

Syntax `SENSe:DATA:TELecom:MEASure:STESts:ENABle <stests enable>`

Parameters

<stests enable> (boolean)	description
OFF or 0	No evaluation (default)
ON or 1	Measurements evaluated

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:MEASURE:STESTS:ENABLE ON`

Related Commands None

SENSe:DATA:TELecom:MEASure:STESts:ENABle?

This query returns the current setting of the pass/fail tests enable.

Syntax `SENSe:DATA:TELecom:MEASure:STESts:ENABle?`

Response

<stests enable> (boolean)	description
0	No evaluation (OFF) (default)
1	Measurements evaluated (ON)

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:MEASURE:STESTS:ENABLE?`

Response: 0

Related Commands SENSe:DATA:TELecom:MEASure:STESts:ENABle

SENSe:DATA:TELecom:MEASure:STESts:DESCription

This command sets the pass/fail test description. This description is stored on the disk with the pass/fail test. The start and end prompts appear in the SAVE PASS/FAIL TESTS menu.

Syntax `SENSe:DATA:TELecom:MEASure:STESts:DESCription <description>, <start prompt>,<end prompt>`

Parameters

<description> (string)	description
An ASCII string, maximum length of 25 bytes	The pass/fail test description
<start prompt> (string)	**description**
An ASCII string, maximum length of 75 bytes	Text to prompt the operator at the start of the test

(continued on next page)

<end prompt> (string)	description
An ASCII string, maximum length of 75 bytes	Text to prompt the operator at the end of the test

Dependencies None

Errors and Events None

Examples

```
SENSE:DATA:TELECOM:MEASURE:STESTS:DESCRIPTION "THIS TESTS AN XYZ
ADM","CONNECT TO EQUIPMENT","REMEMBER TO DISCONNECT FROM
EQUIPMENT"
```

Related Commands None

SENSe:DATA:TELecom:MEASure:STESts:DESCription?

This query returns the pass/fail test description. This description is stored on the disk with the pass/fail test. The start and end prompts appear in the SAVE PASS/FAIL TESTS menu.

Syntax

```
SENSe:DATA:TELecom:MEASure:STESts:DESCription?
```

Response

<description> (string)	description
An ASCII string, maximum length of 25 bytes	The pass/fail test description
<start prompt> (string)	**description**
An ASCII string, maximum length of 75 bytes	Text to prompt the operator at the start of the test
<end prompt> (string)	**description**
An ASCII string, maximum length of 75 bytes	Text to prompt the operator at the end of the test

Dependencies None

Errors and Events None

Examples

Query: `SENSE:DATA:TELECOM:MEASURE:STESTS:DESCRIPTION?`

Response: `"THIS TESTS AN XYZ ADM","CONNECT TO EQUIPMENT", "REMEMBER TO DISCONNECT FROM EQUIPMENT"`

Related Commands SENSe:DATA:TELecom:MEASure:STESts:DESCription

SENSe:DATA:TELecom:MEASure:STESts:THREshold

This command sets the pass/fail test criteria. The pass/fail result is determined by applying the criteria given by this command after the current test is completed. You can set up a maximum of four sets of pass/fail criteria.

Syntax `SENSe:DATA:TELecom:MEASure:STESts:THREshold <criteria number>, <type>,<source>,<threshold>`

Parameters

<criteria number> (discrete)	description
1	First set of pass/fail criteria
2	Second set of pass/fail criteria
3	Third set of pass/fail criteria
4	Fourth set of pass/fail criteria
<type> (discrete)	**description**
NONE	No pass/fail criteria
ALARm	Alarms are the pass/fail criteria
FAILure	Failures are the pass/fail criteria
ERATio	Bit error ratio is the pass/fail criterion
ECOUnt	Error count is the pass/fail criterion
ESEConds	Errored seconds are the pass/fail criteria
POINter	Pointer movements are the pass/fail criteria

if <type> = NONE

<source> (discrete)	**description**
NONE	No pass/fail criteria
<threshold> (discrete)	**description**
0	No threshold

if <type> = ALARm

<source> (discrete)	**description**
ANY	Any alarm
LAIS	Line alarm indication signal
LFERf	Line FERF
PAIS	Path AIS
PFERf	Path FERF
AIS	DS1/DS3 AIS (Add/Drop/Test Option Only)
YELlow	DS1/DS3 AIS (Add/Drop/Test Option Only)
VTAIS	VT AIS (Add/Drop/Test Option Only)
VTFErf	VT FERF (Add/Drop/Test Option Only)
<threshold> (discrete)	**description**
DETected	Threshold is detected
NDETected	Threshold is not detected

if <type> = FAILure

<source> (discrete)	**description**
ANY	Any of the failures below
LOSignal	Loss of Signal
LOFrame	Loss of Frame
LOPointer	STS Loss of Pointer
LOPS	DS1/DS3 loss of pattern sync (Add/Drop/Test Option Only)
VTLOPointer	VT Loss of Pointer (Add/Drop/Test Option Only)

(continued on next page)

if <type> = FAILure

<source> (discrete)	description
VTLOM	VT Loss of Multiframe (Add/Drop/Test Option Only)
<threshold> (discrete)	**description**
DETected	Threshold is detected
NDETected	Threshold is not detected

if <type> = ERATio, ECOUnt, or ESEConds

<source> (discrete)	description
ANY	Any of the errors below
SCV	B1 error
LCV	B2 error
PCV	B3 error
BIT	BIT error
CRC	Cyclic redundancy check (DS1 ESF only)
PARITY	Parity error (DS3 CBIT or M13 only)
VTBIP	VT BIP (Add/Drop/Test Option Only)
VTFEBe	VT far end block error (Add/Drop/Test Option Only)
<threshold> (NR3-numeric)	**description**
Any number	The test will fail for any level greater than this value

if <type> = POINter

<source> (discrete)	description
SPENdf	STS new data flag
SPEJust	STS pointer justification
VTNDf	VT new data flag (Add/Drop/Test Option Only)
VTJUst	VT pointer justification (Add/Drop/Test Option Only)

(continued on next page)

if <type> = POINter

<threshold> (NR3-numeric)	description
Any number	The test will fail for any level greater than this value

Dependencies SENSe:DATA:TELecom:STESts:ENABle must be set to ON for this command to apply.

Errors and Events None

Examples

```
SENSE:DATA:TELECOM:MEASURE:STESTS:THRESHOLD 1,ALARM,ANY,DETECTED
```

Related Commands SENSe:DATA:TELecom:STESts:ENABle

SENSe:DATA:TELecom:MEASure:STESts:THREshold?

This query returns the pass/fail test criteria. The pass/fail result is determined by applying the criteria given by this command after the current test is completed. You can set up a maximum of four sets of pass/fail criteria.

Syntax

```
SENSe:DATA:TELecom:MEASure:STESts:THREshold? <criteria number>
```

Parameters

<criteria number> (discrete)	description
1	First set of pass/fail criteria
2	Second set of pass/fail criteria
3	Third set of pass/fail criteria
4	Fourth set of pass/fail criteria

Response

<type> (discrete)	description
NONE	No pass/fail criteria
ALARm	Alarms are the pass/fail criteria
FAILure	Failures are the pass/fail criteria

(continued on next page)

<type> (discrete)	description
ERATio	Bit error ratio is the pass/fail criterion
ECOUnt	Error count is the pass/fail criterion
ESEConds	Errored seconds are the pass/fail criteria
POINter	Pointer movements are the pass/fail criteria

if <type> = NONE

<source> (discrete)	description
NONE	No pass/fail criteria
<threshold> (discrete)	**description**
0	No threshold

if <type> = ALARm

<source> (discrete)	description
ANY	Any alarm
LAIS	Line alarm indication signal
LFERf	Line FERF
PAIS	Path AIS
PFERf	Path FERF
AIS	DS1/DS3 AIS (Add/Drop/Test Option Only)
YELlow	DS1/DS3 AIS (Add/Drop/Test Option Only)
VTAIS	VT AIS (Add/Drop/Test Option Only)
VTFErf	VT FERF (Add/Drop/Test Option Only)
<threshold> (discrete)	**description**
DETected	Threshold is detected
NDETected	Threshold is not detected

if <type> = FAILure

<source> (discrete)	**description**
ANY	Any of the failures below
LOSignal	Loss of Signal
LOFrame	Loss of Frame
LOPointer	STS Loss of Pointer
LOPS	DS1/DS3 loss of pattern sync (Add/Drop/Test Option Only)
VTLOPointer	VT Loss of Pointer (Add/Drop/Test Option Only)
VTLOM	VT Loss of Multiframe (Add/Drop/Test Option Only)
<threshold> (discrete)	**description**
DETected	Threshold is detected
NDETected	Threshold is not detected

if <type> = ERATio, ECOUnt, or ESEConds

<source> (discrete)	**description**
ANY	Any of the errors below
SCV	B1 error
LCV	B2 error
PCV	B3 error
BIT	BIT error
CRC	Cyclic redundancy check (DS1 ESF only)
PARITY	Parity error (DS3 CBIT or M13 only)
VTBIP	VT BIP (Add/Drop/Test Option Only)
VTFEBe	VT far end block error (Add/Drop/Test Option Only)
<threshold> (NR3-numeric)	**description**
Any number	The test will fail for any level greater than this value

if <type> = POINter

<source> (discrete)	description
SPENdf	STS new data flag
SPEJust	STS pointer justification
VTNDf	VT new data flag (Add/Drop/Test Option Only)
VTJUst	VT pointer justification (Add/Drop/Test Option Only)
<threshold> (NR3-numeric)	**description**
Any number	The test will fail for any level greater than this value

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:MEASURE:STESTS:THRESHOLD? 1`

Response: `ALARM,ANY,DETECTED`

Related Commands SENSe:DATA:TELecom:MEASure:STESts:THREshold

SENSe:DATA:TELecom:MEASure:STESts:RESUlts?

This query returns the results of applying the pass/fail criteria to the pass/fail test measurements.

Syntax `SENSe:DATA:TELecom:MEASure:STESts:RESUlts?`

Response

<test results> (discrete)	description
NONE	Test is still running or no pass/fail test measurements have been requested
PASSED	Test passed
FAILED	Test failed

Dependencies The test must be completed for the results to be valid.

Errors and Events 200, "Execution error; Results not available"

Examples Query: `SENSE:DATA:TELECOM:MEASURE:STESTS:RESULTS?`

Response: `PASSED`

Related Commands SENSe:DATA:TELecom:MEASure:STESts:THREshold

SENSe:DATA:TELecom:MEASure:STESts:DONE

This command specifies what action the instrument takes when the pass/fail test calculations complete.

NOTE. *The pass/fail test results are written to the file name specified by the MMEMory:STORe:TESTs command. So, if you run consecutive pass/fail tests, remember to send the MMEMory:STORe:TESTs command with unique file names in between each pass/fail test run so the pass/fail test results files are not overwritten.*

Syntax `SENSe:DATA:TELecom:MEASure:STESts:DONE <action>`

Parameters

<action> (discrete)	description
NONE	Take no action when pass/fail tests calculations complete
PRINT	Print the pass/fail tests results
DISK	Save the pass/fail tests results to disk; the file name is specified by the MMEMory:STORe:TESTs command

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:MEASURE:STESTS:DONE NONE`

Related Commands SENSe:DATA:TELecom:MEASure:STESTs:ENABle
SENSe:DATA:TELecom:MEASure:STESTs:THREshold
MMEMory:STORe:TESTs

SENSe:DATA:TELecom:MEASure:STESts:DONE?

This query returns what action the instrument takes when the pass/fail test calculations complete.

Syntax `SENSe:DATA:TELecom:MEASure:STESts:DONE?`

Response

<action> (discrete)	description
NONE	Take no action when pass/fail tests calculations complete
PRINT	Print the pass/fail tests results
DISK	Save the pass/fail tests results to disk; the file name is specified by the MMEMory:STORe:TESTs command

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:MEASURE:STESTS:DONE?`

Response: `NONE`

Related Commands SENSe:DATA:TELecom:MEASure:STESTs:DONE

SENSe:DATA:TELecom:AUTOscan Subsystem

This section describes the command that automatically configures the receiver to the attached signal on any of the input connectors. Figure 2–35 shows the hierarchy tree for this subsystem.

Figure 2–35: SENSe:DATA:TELecom:AUTOscan Subsystem

SENSe:DATA:TELecom:AUTOscan

This command starts the autoscan function which sets up the receiver based on the connected signal. The Operation Complete bit is set when this command has completed. Use the SYSTem:ERRor? query to see if the autoscan completed successfully.

Syntax `SENSe:DATA:TELecom:AUTOscan`

Parameters None

Dependencies A valid signal should be connected.

Errors and Events 361, “Autoscan failed; Instrument returned to previous setup”
361, “Autoscan failed; Autoscan already in progress”
402, “Operation complete; Autoscan complete”
200, “Execution error; Autoscan incomplete — no signals connected”

Examples `SENSE:DATA:TELECOM:AUTOSCAN`

Related Commands SYSTem:ERRor?

SENSe:DATA:TELecom:TRIButary Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that allow you to view a dropped tributary signal. Figure 2–35 shows the hierarchy tree for this subsystem.

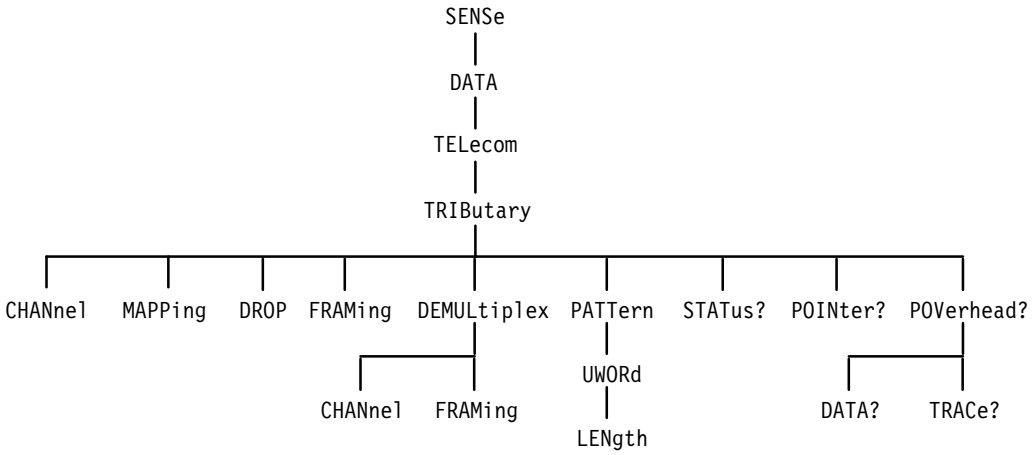


Figure 2–36: SENSe:DATA:TELecom:TRIButary Subsystem

SENSe:DATA:TELecom:TRIButary:DROP

Add/Drop/Test Option Only

This command enables the tributary drop capability. If enabled, the demapped signal is available on the output connector specified by mapping. DS3_DEMUX mapping drops a DS3 signal.

Syntax `SENSe:DATA:TELecom:TRIButary:DROP <trib drop>`

Parameters

<trib drop> (boolean)	description
OFF or 0	Tributary signal not available on output connector (default)
ON or 1	Tributary signal available on output connector

Dependencies SENSe:DATA:TELecom:SOURce must be set to INPUT1.
SENSe:DATA:TELecom:MAPPing must be set to TRIButary.

Errors and Events 221, “Settings conflict; Instrument unable to drop signal while transmitting current rate”
221, “Settings conflict; Instrument unable to drop while transmitter not externally adding 140Mb”

Examples

```
SENSE:DATA:TELECOM:TRIBUTARY:DROP ON
```

Related Commands SOURce:DATA:TELecom:SOURce
SENSe:DATA:TELecom:SOURce
SENSe:DATA:TELecom:MAPPing

SENSe:DATA:TELecom:TRIButary:DROP?

Add/Drop/Test Option Only

This query returns the tributary drop status.

Syntax

```
SENSe:DATA:TELecom:TRIButary:DROP?
```

Response

<trib drop> (boolean)	description
0	Tributary signal not available on output connector (default)
1	Tributary signal available on output connector

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:DROP?`

Response: 1

Related Commands SENSe:DATA:TELecom:TRIButary:DROP

SENSe:DATA:TELecom:TRIButary:CHANnel

Add/Drop/Test Option Only

This command selects the VTASYNC channel. Use the SENSe:DATA:TELecom:TRIButary:MAPPing command to set the tributary demapping.

Syntax `SENSe:DATA:TELecom:TRIButary:CHANnel <trib channel>`

Parameters

<trib channel> (NR1-numeric)	description
Any integer between 1 and 28	VTASYNC mapping for the active channel (default = 1)
1	DS3 mapping for the active channel

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:TRIBUTARY:CHANNEL 1`

Related Commands SENSe:DATA:TELecom:TRIButary:MAPPing

SENSe:DATA:TELecom:TRIButary:CHANnel?

Add/Drop/Test Option Only

This returns the current tributary channel.

Syntax `SENSe:DATA:TELecom:TRIButary:CHANnel?`

Response

<trib channel> (NR1-numeric)	description
Any integer between 1 and 28	VTASYNC mapping for the active channel (default = 1)
1	DS3 mapping for the active channel

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:CHANNEL?`

Response: 1

Related Commands SENSe:DATA:TELecom:TRIButary:CHANnel

SENSe:DATA:TELecom:TRIButary:MAPPing

Add/Drop/Test Option Only

This command selects the tributary payload demapping. When you are actively mapping and demapping a tributary signal, the SOURce:DATA:TELecom:TRIButary:MAPPing and SENSe:DATA:TELecom:TRIButary:MAPPing functions are coupled; a change to one causes the same change to the other.

Syntax `SENSe:DATA:TELecom:TRIButary:MAPPing <trib mapping>`

Parameters

<trib mapping> (discrete)	description
VTASYNC	Demapped DS1 signal into a VTASYNC (default)
DS3	Demapped DS3 signal
DS3_DEMUX	DS3 demultiplex DS1

Dependencies None

Errors and Events None

Examples

```
SENSE:DATA:TELECOM:TRIBUTARY:MAPPING VTASYNC
```

Related Commands

SOURce:DATA:TELecom:TRIButary:MAPPing
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel

SENSe:DATA:TELecom:TRIButary:MAPPing?

Add/Drop/Test Option Only

This query returns the current tributary payload demapping.

Syntax

```
SENSe:DATA:TELecom:TRIButary:MAPPing?
```

Response

<trib mapping> (discrete)	description
VTASYNC	Demapped DS1 signal into a VTASYNC (default)
DS3	Demapped DS3 signal
DS3_DEMUX	DS3 demultiplex DS1

Dependencies

None

Errors and Events

None

Examples

Query:

```
SENSE:DATA:TELECOM:TRIBUTARY:MAPPING?
```

Response:

```
VTASYNC
```

Related Commands

SENSe:DATA:TELecom:TRIButary:MAPPing
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel

SENSe:DATA:TELecom:TRIButary:FRAMing

Add/Drop/Test Option Only

This command selects the framing of the received tributary signal.

Syntax `SENSe:DATA:TELecom:TRIButary:FRAMing <trib framing>`

Parameters

<trib framing> (discrete)	description
UNFRamed	No framing (default)
SF	DS1 superframe
ESF	DS1 extended superframe
CBIT	DS3 C–bit format
M13	DS3 M13 format

Dependencies For DS3_DEMUX mapping only CBIT and M13 are allowed

Errors and Events None

Examples `SENSE:DATA:TELECOM:TRIBUTARY:FRAMING UNFRAMED`

Related Commands None

SENSe:DATA:TELecom:TRIButary:FRAMing?

Add/Drop/Test Option Only

This query returns the current framing option of the received tributary signal.

Syntax `SENSe:DATA:TELecom:TRIButary:FRAMing?`

Response

<trib framing> (discrete)	description
UNFRamed	No framing (default)
SF	DS1 superframe
ESF	DS1 extended superframe

(continued on next page)

<trib framing> (discrete)	description
CBIT	DS3 C–bit format
M13	DS3 M13 format

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:FRAMING?`

Response: `UNFRAMED`

Related Commands SENSe:DATA:TELecom:TRIButary:FRAMing

SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel

Add/Drop/Test Option Only

This command selects the DS1 channel demultiplexed from a DS3 signal. *RST selects channel 1.

Syntax `SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel <ds1 channel>`

Parameters

<DS1 channel> (discrete)	description
1 – 28	DS1 channel

Dependencies None

Errors and Events None

Examples `SENSE:DATA:TELECOM:TRIBUTARY:DEMULTIPLEX:CHANNEL 1`

Related Commands SENSe:DATA:TELecom:TRIButary:MAPPing
*RST

SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel?

Add/Drop/Test Option Only

This query returns the current DS1 channel being demultiplexed from a DS3 signal.

Syntax `SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel?`

Response

<DS1 channel> (discrete)	description
1 – 28	DS1 channel

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:DEMULTIPLEX:CHANNEL?`

Response: 1

Related Commands SENSe:DATA:TELecom:TRIButary:MAPPing

SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing

Add/Drop/Test Option Only

This command selects the framing of the DS1 signal demultiplexed from a DS3 signal.

Syntax `SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing <trib framing>`

Parameters

<DS1 trib framing> (discrete)	description
UNFRamed	(default)
SF	SUPERFRAME
ESF	Extended SUPERFRAME

Dependencies DS3_DEMUX mapping must be selected

Errors and Events None

Examples `SENSE:DATA:TELECOM:TRIBUTARY:DEMULTIPLEX:FRAMING UNFRAMED`

Related Commands SENSe:DATA:TELecom:TRIButary:MAPPing

SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing?

Add/Drop/Test Option Only

This query returns the framing of the DS1 signal being demultiplexed from a DS3 signal.

Syntax `SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing?`

Response

<DS1 trib framing> (discrete)	description
UNFRamed	(default)
SF	SUPERFRAME
ESF	Extended SUPERFRAME

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:DEMULTIPLEX:FRAMING?`

Response: `UNFRAMED`

Related Commands SENSe:DATA:TELecom:TRIButary:FRAMing

SENSe:DATA:TELecom:TRIButary:PATTern

Add/Drop/Test Option Only

This command selects the internally generated pattern that is placed in the tributary payload.

Syntax

```
SENSe:DATA:TELecom:TRIButary:PATTern <trib pattern>
```

Parameters

<trib pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 is placed in the tributary payload (default)
QRSS	Quasi-random signal source pattern; DS1 rate only
PRBS15	A pseudo-random binary sequence of length 2^{15}–1 is placed in the tributary payload
PRBS20	A pseudo-random binary sequence of length 2^{20}–1 is placed in the tributary payload
AZEROs	All zeros are placed in the payload
AONEs	All ones are placed in the payload
UWORd	A user-defined pattern is placed in the payload
UNKNown	Unknown pattern
FIXED_1_8	1 bit in 8 (DS1, Add/Drop/Test Option Only)
FIXED_3_24	3 bits in 24 (DS1, Add/Drop/Test Option Only)

Dependencies

None

Errors and Events

None

Examples

```
SENSE:DATA:TELECOM:TRIBUTARY:PATTERN PRBS15
```

Related Commands

None

SENSe:DATA:TELecom:TRIButary:PATTern?

Add/Drop/Test Option Only

This query returns the current internally generated tributary payload pattern.

Syntax `SENSe:DATA:TELecom:TRIButary:PATTern?`

Response

<trib pattern> (discrete)	description
PRBS23	A pseudo-random binary sequence of length 2^{23}–1 is placed in the tributary payload (default)
QRSS	Quasi-random signal source pattern; DS1 rate only
PRBS15	A pseudo-random binary sequence of length 2^{15}–1 is placed in the tributary payload
PRBS20	A pseudo-random binary sequence of length 2^{20}–1 is placed in the tributary payload
AZEROs	All zeros are placed in the payload
AONEs	All ones are placed in the payload
UWORd	A user-defined pattern is placed in the payload
UNKNown	Unknown pattern
FIXED_1_8	1 bit in 8 (DS1, Add/Drop/Test Option Only)
FIXED_3_24	3 bits in 24 (DS1, Add/Drop/Test Option Only)

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:PATTERN?`

Response: `AONES`

Related Commands SENSe:DATA:TELecom:TRIButary:PATTern

SENSe:DATA:TELecom:TRIButary:PATTern:UWORd

Add/Drop/Test Option Only

This command sets the user-defined pattern that is placed in the tributary payload.

Syntax

```
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd <trib user pattern>
```

Parameters

<trib user pattern> (hexadecimal)	description
Any 8, 16, or 24 bit hexadecimal number in the range #H00 to #HFFFFFF	Repeating pattern is placed in the tributary payload (default = #H00)

Dependencies

SENSe:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this command to apply. Use the SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth command to set the length of the repeating pattern.

Errors and Events

None

Examples

```
SENSE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD #HAA5500
```

Related Commands

SENSe:DATA:TELecom:TRIButary:PATTern
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth

SENSe:DATA:TELecom:TRIButary:PATTern:UWORd?

Add/Drop/Test Option Only

This query returns the user-defined pattern that is placed in the tributary payload.

Syntax

```
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd?
```

Response

<trib user pattern> (hexadecimal)	description
Any 8, 16, or 24 bit hexadecimal number in the range #H00 to #HFFFFFF	Repeating pattern is placed in the tributary payload (default = #H00)

Dependencies SENSe:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this query to apply.

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD?`

Response: `#HAA5500`

Related Commands SENSe:DATA:TELecom:TRIButary:PATTern:UWORd

SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth

Add/Drop/Test Option Only

This command sets the number of bytes of the user-defined pattern that are repeated in the tributary payload.

Syntax `SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth <trib user pattern length>`

Parameters

<trib user pattern length> (NR1-numeric)	description
Any integer in the range 1 to 3	Number of bytes of user-defined pattern that are repeated in the tributary payload (default = 1)

Dependencies SENSe:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this command to apply. Use the SENSe:DATA:TELecom:TRIButary:PATTern: UWORd command to set the repeating pattern.

Errors and Events None

Examples `SENSE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD:LENGTH 3`

Related Commands SENSe:DATA:TELecom:TRIButary:PATTern
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd

SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth?

Add/Drop/Test Option Only

This query returns the number of bytes of the user-defined pattern that are repeated in the tributary payload.

Syntax

```
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth?
```

Response

<trib user pattern length> (NR1-numeric)	description
Any integer in the range 1 to 3	Number of bytes of user-defined pattern that are repeated in the tributary payload (default = 1)

Dependencies

SENSe:DATA:TELecom:TRIButary:PATTern must be set to UWORd for this query to apply.

Errors and Events

None

Examples

Query: `SENSE:DATA:TELECOM:TRIBUTARY:PATTERN:UWORD:LENGTH?`

Response: 3

Related Commands

SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth

SENSe:DATA:TELecom:TRIButary:POINter?

Add/Drop/Test Option Only

This query returns the current tributary pointer value.

Syntax

```
SENSe:DATA:TELecom:TRIButary:POINter?
```

Response

<trib pointer> (NR1-numeric)	description
Any integer in the range 0 to 1023	Tributary pointer value for VTASYNC mapping (default = 78, illegal > 103)

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:POINTER?`

Response: `12`

Related Commands None

SENSe:DATA:TELecom:TRIButary:STATus?

Add/Drop/Test Option Only

This query returns the status of the received tributary signal. The returned status is not an accumulated status; the response reflects only the status of the tributary signal at the time the query is sent.

Syntax `SENSe:DATA:TELecom:TRIButary:STATus?`

Parameters

<decimal value> (NR1-numeric)	bit	definition
1	0	INPUT2/INPUT3 LOS
2	1	INPUT2/INPUT3 LOF
4	2	Unused
8	3	Unused
16	4	VT LOP
32	5	VT AIS
64	6	DS1/DS3 AIS
128	7	DS1/DS3 Yellow
256	8	Frame error
512	9	DS1/DS3 error
1024	10	VT FERF
2048	11	VT pointer adjustment
4096	12	VT NDF
8192	13	Pattern lock

(continued on next page)

<decimal value> (NR1-numeric)	bit	definition
16384	14	DS3 idle
32768	15	VT LOM

Dependencies None

Errors and Events None

Examples Query: `SENSE:DATA:TELECOM:TRIBUTARY:STATUS?`

Response: 32

Related Commands SENSe:DATA:TELecom:TRIButary:STATus

SENSe:DATA:TELecom:TRIButary:POVerhead:DATA?

This query returns the value in the specified VT1.5 path overhead byte.

Syntax `SENSe:DATA:TELecom:TRIButary:POVerhead:DATA? <byte name>`

Parameters

<byte name> (discrete)	description
V5	(VT1.5)

Response

<value> (NR1-numeric)	description
Any number in the range 0 to 255	The byte is set to this value (the value for J1 is the ASCII representation of the string value)

Dependencies Use the SENSe:DATA:TELecom:TRIButary:CHANNel command to specify which path trace to query. INITiate and TRIGGer:IMMediate affect these bytes.

Errors and Events None

Examples

Query: `SENSE:DATA:TELECOM:TRIBUTARY:POVERHEAD:DATA? V5`

Response: `123`

Related Commands

INITiate
TRIGger:IMMediate

SENSe:DATA:TELecom:MEASure:TRIButary Subsystem

Add/Drop/Test Option Only

This section describes the commands and queries that access tributary error, alarm, failure, and pointer measurements for current and previous tests. Figures 2–37, 2–38, and 2–39 show the hierarchy trees for this subsystem.

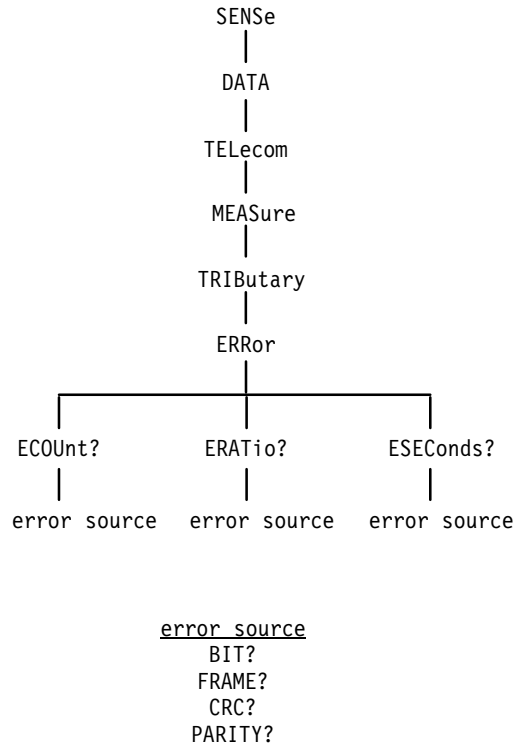
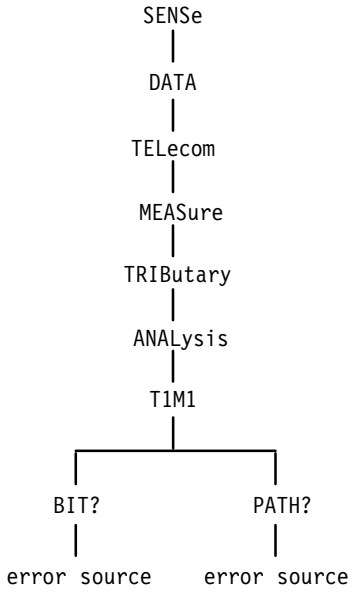


Figure 2–37: SENSe:DATA:TELecom:MEASure:TRIButary:ERRor Subsystem (DS1/DS3)



error_source
ECOUnt?
ESEConds?
PESeconds?
AESeconds?
PAESeconds?
BESeconds?
PBESeconds?
UASeconds?
PUASeconds?
EFSeconds?
PEFSeconds?
SESeconds?
PSESeconds?

Figure 2–38: SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis Subsystem (DS1/DS3)

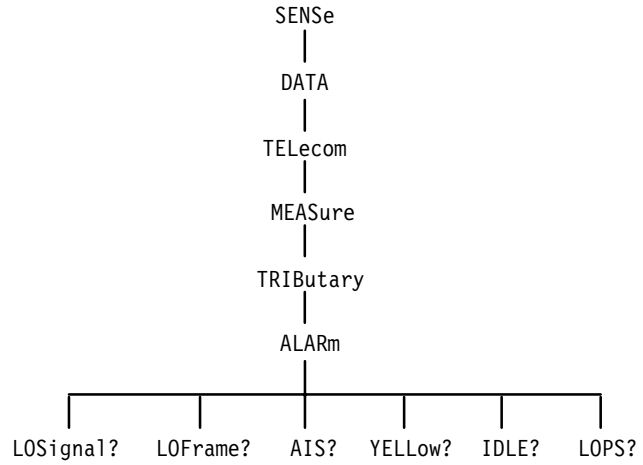


Figure 2–39: SENSe:DATA:TELecom:MEASure:TRIButary:ALARm Subsystems (DS1/DS3)

A variety of tributary error, alarm, and failure measurements are reported through this subsystem. Table 2–25 shows how error, alarm, and failure measurements are calculated. Tables 2–26 and 2–27 show how the analysis measurements are calculated. These calculations are based on T1M1.93 specifications.

Table 2–25: How Error and Alarm Measurements are Calculated

Type of Measurement	Method of Calculation
Error count	Number of bit errors that were errored in the signal
Bit Error Ratio (BER)	Ratio of error count to the total number of received bits
Errored seconds	Number of seconds that had any error counts
Alarms	Number of one-second intervals that contained a specific alarm such as Loss of Signal (LOS) and Loss of Frame (LOF)

Table 2–26: How Analysis Measurements are Calculated

Type of Measurement	Method of Calculation
Error count	Number of bit errors not occurring during periods of unavailability (see Unavailable seconds)
Errored seconds	Total number of seconds that had any error count; does not include any period of unavailability (see Unavailable seconds)
Type A errored seconds	Number of seconds that had exactly one error count

Table 2–26: How Analysis Measurements are Calculated (Cont.)

Type of Measurement	Method of Calculation
Type B errored seconds	Number of seconds that had more than one error count and less than N errors (see Table 2–27)
Severely errored seconds	Number of seconds with more than N errors (see Table 2–27)
Unavailable seconds	Number of seconds that the signal had too many errors to be available for use; unavailability starts at the onset of ten contiguous severely errored seconds
Error free seconds	Number of seconds that contained zero errors

Table 2–27: Value of N for Analysis Measurements (DS1/DS3)

Tributary Rate	Framing	Type of Error	N
DS1	SF (superframe)	Frame error	8
	ESF (extended super-frame)	CRC	320
DS3	M13	Parity (P-bit parity)	45
	CBIT	Parity (C-bit parity)	45

Some of queries in this section have their information presented in a way that is different from queries in the rest of the manual. The syntax and examples are in table format. Figure 2–40 shows you how to read the Syntax Tables in this section. Follow the step numbers to create any query. Table 2–28 explains the terms used in the SENSe:DATA:TELecom:MEASure Syntax Tables.

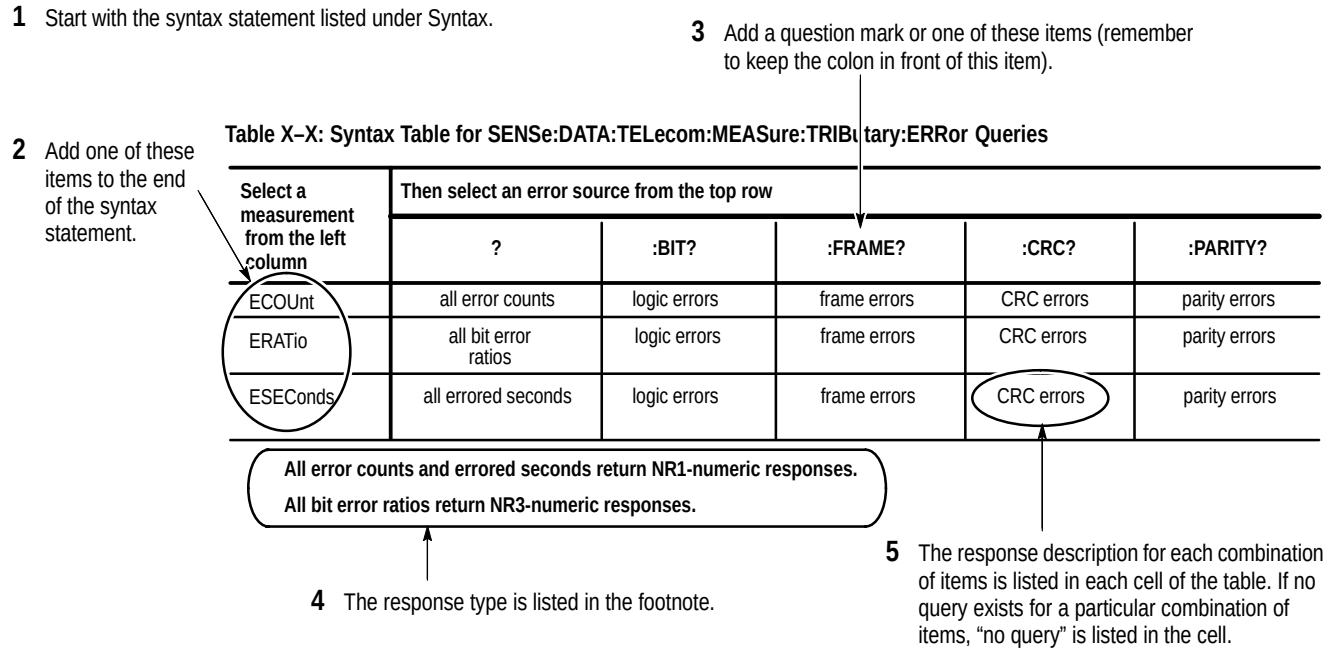


Figure 2–40: How to Read the Syntax Tables in the SENSe:DATA:TELecom:MEASure:TRIButary Subsystem Section

Table 2–28: Terms Used in the SENSe:DATA: TELecom:MEASure:TRIButary Queries

Term	Meaning
BIT	Pattern bit error
FRAME	Frame bit error
CRC	CRC error
PARITY	Parity error

Figure 2–41 shows you how to read the Example Tables in this section.

Table X–X: Example Table for SENSe:DATA:TELecom:MEASure:TRIButary:ERRor:Queries

Query	Response
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ERROR:ECOUNT:BIT?	714
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ERROR:ERATIO:FRAME?	1.0E-8
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ERROR:ESECONDS:CRC?	3

Selected examples of queries are shown in the left column

A typical response is shown in the right column for each example

Figure 2–41: How to Read the Example Tables in the SENSe:DATA:TELecom:MEASure:TRIButary Subsystem Section

SENSe:DATA:TELecom:MEASure:TRIButary:ERRor Queries

Add/Drop/Test Option Only

These queries return tributary error measurements. When you use the high-level queries (for example, SENSe:DATA:TELecom:MEASure:TRIButary:ERRor? or SENSe:DATA:TELecom:MEASure:TRIButary:ERRor:ECOUnt?), it is helpful to turn the headers on (SYSTem:HEADers ON) so you can identify each response value in the response string.

Syntax `SENSe:DATA:TELecom:MEASure:TRIButary:ERRor?`

`SENSe:DATA:TELecom:MEASure:TRIButary:ERRor:[measurement]:[error source]` (see Table 2–29 to complete the query)

Table 2–29: Syntax Table for SENSe:DATA:TELecom:MEASure:TRIButary:ERRor Queries (DS1/DS3)

Select a measurement from the left column	Then select an error source from the top row				
	?	**:BIT?**	**:FRAME?[1]**	**:CRC?[2]**	**:PARITY?[3]**
ECOUnt	all error counts	logic errors	frame errors	CRC errors	parity errors
ERATio	all bit error ratios	logic errors	frame errors	CRC errors	parity errors
ESEConds	all errored seconds	logic errors	frame errors	CRC errors	parity errors

All error counts and errored seconds return NR1-numeric responses.

All bit error ratios return NR3-numeric responses.

1 **The FRAME error source is valid only for DS1/DS3 framed signals.**

2 **The CRC error source is valid only for DS1 rate and ESF framing.**

3 **The PARITY error source is valid only for DS3 rate and CBIT or M13 framing.**

Response See Table 2–29.

Dependencies These measurement queries can be sent at any time. But, if a test is currently running, the responses to the queries might not represent the final error measurements. After a test has been stopped or the test duration has expired, you can send these measurement queries again to get the final error measurements.

Do not set SENSe:DATA:TELecom:TRIButary:PAYLoad:PATTern to UNKNown if you want to use these queries.

Errors and Events None

Examples See Table 2–30.

Table 2–30: Example Table for SENSe:DATA:TELecom:MEASure:TRIButary:ERRor Queries

Query	Response
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ERROR:ECOUNT:BIT?	714
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ERROR:ERATIO:FRAME?	1.0E-8
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ERROR:ESECONDS:CRC?	3

Related Commands SENSe:DATA:TELecom:TEST:STARt
SENSe:DATA:TELecom:TEST:STOP
SENSe:DATA:TELecom:TRIButary:PAYLoad:PATTern
SYSTem:HEADers

SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis Queries

Add/Drop/Test Option Only

These queries return an analysis of section, line, path, and payload errors for tributary signals. When you use the high-level queries (for example, SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis? or SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis:ECOUnt?), it is helpful to turn the headers on (SYSTem:HEADers ON) so you can identify each response value in the response string.

Syntax `SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis?`

`SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis:T1M1?`

SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis:T1M1:BIT:[error source] (see Table 2–31 to complete the query)

SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis:T1M1:PATH:[error source] (see Table 2–31 to complete the query)

Table 2–31: Syntax Table for SENSe:DATA:TELecom:MEASure:TRIButary: ANALysis:T1M1:BIT and :PATH Queries (DS1/DS3)

Select one of these error sources	Response
?	All T1M1 payload bit analysis responses below
ECOUnt	Error count
ESECOnds	Errored seconds
PESeconds	Percent errored seconds
AESeconds	Type A errored seconds
PAESeconds	Percent Type A errored seconds
BESeconds	Type B errored seconds
PBESeconds	Percent Type B errored seconds
UASeconds	Unavailable seconds
PUASeconds	Percent unavailable seconds
EFSeconds	Error free seconds
PEFSeconds	Percent error free seconds
SESeconds	Severely errored seconds
PSESeconds	Percent severely errored seconds

All percent measurements return NR3-numeric responses.

All other measurements return NR1-numeric responses.

Response

See Table 2–31.

Dependencies

These measurement queries can be sent at any time. However, if a test is currently running, the responses to the queries might not represent the final error measurements. After a test has been stopped or the test duration has expired, you can send these measurement queries again to get the final error measurements.

Do not set SENSe:DATA:TELecom:TRIButary:PAYLoad:PATTern to UNKNown if you want to use these queries.

Errors and Events

None

Examples See Table 2–32.

Table 2–32: Example Table for SENSe:DATA:TELecom:MEASure:TRIButary:ANALysis Queries (DS1/DS3)

Query	Response
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ANALYSIS:T1M1:BIT:AESECONDS?	76824
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ANALYSIS:T1M1:PATH:PEFSECONDS?	6.5E-3

Related Commands SENSe:DATA:TELecom:TEST:STARt
SENSe:DATA:TELecom:TEST:STOP
SENSe:DATA:TELecom:TRIButary:PAYLoad:PATTern
SYSTem:HEADers

SENSe:DATA:TELecom:MEASure:TRIButary:ALARm Queries

Add/Drop/Test Option Only

These queries return tributary alarm measurements. When you use the SENSe:DATA:TELecom:MEASure:TRIButary:ALARm? query, it is helpful to turn the headers on (SYSTem:HEADers ON) so you can identify each response value in the response string.

Syntax All valid queries are listed in the Syntax column of Table 2–33.

Table 2–33: Syntax Table for SENSe:DATA:TELecom:MEASure:TRIButary:ALARm Queries

Syntax	Response
SENSe:DATA:TELecom:MEASure:TRIButary:ALARm?	All tributary alarm measurements
SENSe:DATA:TELecom:MEASure:TRIButary:ALARm:LOSignal?	Number of seconds of tributary Loss of Signal
SENSe:DATA:TELecom:MEASure:TRIButary:ALARm:LOFrame?	Number of seconds of tributary Loss of Frame
SENSe:DATA:TELecom:MEASure:TRIButary:ALARm:AIS?	Number of seconds of tributary AIS
SENSe:DATA:TELecom:MEASure:TRIButary:ALARm:YELLow?	Number of seconds of tributary Path Yellow (DS1/DS3 only)
SENSe:DATA:TELecom:MEASure:TRIButary:ALARm:IDLE?	Number of seconds of tributary idle (DS3 only)
SENSe:DATA:TELecom:MEASure:TRIButary:ALARm:LOPS?	Number of seconds of loss of pattern sync

All responses are in NR1-numeric format.

Response See the Response column of Table 2–33.

Dependencies These measurement queries can be sent at any time. However, if a test is currently running, the responses to the queries might not represent the final error measurements. After a test has been stopped or the test duration has expired, you can send these measurement queries again to get the final error measurements.

Errors and Events None

Examples See Table 2–34.

Table 2–34: Example Table for SENSe:DATA:TELecom:MEASure:TRIButary:ALARm Queries

Query	Response
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ALARM:LOSIGNAL?	20
SENSE:DATA:TELECOM:MEASURE:TRIBUTARY:ALARM:AIS?	13

Related Commands SENSe:DATA:TELecom:TEST:STARt
SENSe:DATA:TELecom:TEST:STOP
SENSe:DATA:TELecom:TRIButary:PAYLoad:PATTern
SYSTem:HEADers

Transmitter/Receiver Setup Commands

The Transmitter/Receiver Setup Commands allow you to control the interaction between Transmitter and Receiver settings. This section contains all of the commands and queries for the following Transmitter/Receiver Setup subsystem:

- INSTrument

INSTrument Subsystem

This section describes the command and query that control the coupling between the Transmitter and Receiver setups. Figure 2–42 shows the hierarchy tree for this subsystem.

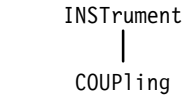


Figure 2–42: INSTrument:COUPling Subsystem

INSTrument:COUPling

This command sets the Transmitter and Receiver setup coupling. NONE allows the Transmitter and Receiver to be independently controlled. TXRX establishes interaction between the Transmitter and Receiver parameters shown in Table 2–35 with the Transmitter governing the *initial* Receiver setup. RXTX establishes interaction between the Transmitter and Receiver parameters shown in Table 2–35 with the Receiver governing the *initial* Transmitter setup.

Table 2–35: Parameters Interacting Through Instrument Coupling

Receiver	Transmitter
INPUT1:RATE	OUTPUT1:RATE
INPUT1:TYPE	OUTPUT1:TYPE
SENSE:DATA:TELecom:SOURce	SOURce:DATA:TELecom:SOURce
SENSe:DATA:TELecom:CHANnel	SOURce:DATA:TELecom:CHAnnel
SENSe:DATA:TELecom:PAYLoad:MAPPing	SOURce:DATA:TELecom:PAYLoad:MAPPing
SENSe:DATA:TELecom:PAYLoad:PATTern	SOURce:DATA:TELecom:PAYLoad:PATTern
SENSe:DATA:TELecom:PAYLoad:UBYTe	SOURce:DATA:TELecom:PAYLoad:UBYTe

NOTE. *A change to one of the parameters listed in Table 2–35 might cause a change to the INSTrument:COUPling parameter value. For example, if you set INSTrument:COUPling to TXRX then change OUTPUT1:RATE, the value of INSTrument:COUPling is changed to RXTX.*

Syntax `INSTrument:COUPling <coupling>`

Parameters

<coupling> (discrete)	description
NONE	Setups are independent
TXRX	Transmitter sets initial condition of the Receiver
RXTX	Receiver sets initial condition of the Transmitter

Dependencies None

Errors and Events None

Examples `INSTRUMENT:COUPLING TXRX`

Related Commands None

INSTrument:COUPling?

This query returns the current setting of the Transmitter and Receiver setup coupling. NONE allows the Transmitter and Receiver to be independently controlled. TXRX establishes interaction between the Transmitter and Receiver parameters listed in Table 2–35 with the Transmitter governing the *initial* Receiver setup. RXTX establishes interaction between certain Transmitter and Receiver parameters listed in Table 2–35 with the Receiver governing the *initial* Transmitter setup.

Syntax `INSTrument:COUPling?`

Response

<coupling> (discrete)	description
NONE	Setups are independent
TXRX	Transmitter sets initial condition of the Receiver
RXTX	Receiver sets initial condition of Transmitter

Dependencies None

Errors and Events None

Examples Query: `INSTRUMENT:COUPLING?`

Response: `TXRX`

Related Commands None

Trigger and Capture Commands

The Trigger and Capture commands allow you to capture overhead and payload data. This section contains all of the commands and queries for each of the following Trigger and Capture subsystems:

- TRIGger

TRIGger Subsystem

This section describes each of the commands and queries that allow you to arm the trigger system and force an immediate trigger. Figure 2–43 shows the hierarchy tree for this subsystem.

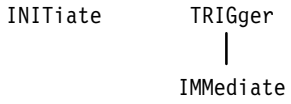


Figure 2–43: TRIGger Subsystem

INITiate

This command causes the capture mechanism to start and the trigger system to arm. After receiving this command, the instrument will acquire data until the programmed trigger event occurs or the capture is stopped manually with the ABORt or TRIGger:IMMediate command.

Syntax `INITiate`

Parameters None

Dependencies None

Errors and Events None

Examples

```
INITIATE
```

Related Commands

TRIGger:IMMediate
SENSe:DATA:TELecom:OVERhead:DATA?
SENSe:DATA:TELecom:POVerhead:DATA?

TRIGger:IMMediate

This command forces a trigger to occur, which stops the capture of data.

Syntax

```
TRIGger:IMMediate
```

Parameters

None

Dependencies

Before the TRIGger:IMMediate command is sent, an INITiate command must be sent first to arm the trigger system.

Errors and Events

None

Examples

```
TRIGGER:IMMEDIATE
```

Related Commands

INITiate
SENSe:DATA:TELecom:OVERhead:DATA?
SENSe:DATA:TELecom:POVerhead:DATA?

Input/Output Commands

The Input/Output commands allow you to write files to disk, control the display of the instrument, print reports, and communicate with the instrument over an RS-232 port. This section contains all of the commands and queries for each of the following Input/Output subsystems:

- MMEMory
- DISPlay
- HCOPy
- SYSTem:COMMunicate:SERial

MMEMory Subsystem

This section describes each of the commands and queries that control access to the disk. Figure 2–44 shows the hierarchy tree for this subsystem. All MMEMory commands set the OPC bit in the Standard Event Status Register.

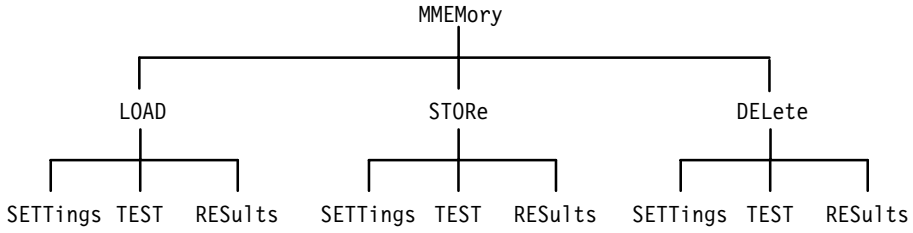


Figure 2–44: MMEMory Subsystem

NOTE*. Do not specify the file name extension for any of these commands. Extensions are added automatically by the system.*

MMEMory:STORe:SETTings

This command stores the current instrument state or stored setups to a disk file.

Syntax `MMEMory:STORe:SETTings <file name>,<buffer>`

Parameters

<file name> (string)	description
A maximum of eight characters in the form "file name"	File name you want to save the setups to; the extension ".SET" is added by the software
<buffer> (NR1-numeric)	**description**
Valid values are 0 through 5	Buffer 0 contains the current setups; buffers 1 through 5 contain the previous setups

Dependencies A disk must be in the disk drive.

Errors and Events
402, "Operation complete; Save of instrument setup complete"
252, "Missing media; Disk not present in drive"
254, "Media full; Disk is full"
257, "File name error; File name required"
258, "Media protected; Disk is write-protected, cannot write to file"

Examples
```
MMEMORY:STORe:SETTings "TEMPOOO1",1
```

Related Commands *SAV

MMEMory:DELete:SETTings

This command deletes a setup file from the disk.

Syntax
```
MMEMory:DELete:SETTings <file name>
```

Parameters

<file name> (string)	description
A maximum of eight characters in the form "file name"	Name of the setup file you want deleted

Dependencies A disk must be in the disk drive.

Errors and Events
252, "Missing media; Disk not present in drive"
256, "File name not found; Could not open disk file"
258, "Media protected; Disk is write-protected, cannot write to file"

Examples

```
MMEMORY:DELETE:SETTINGS "FILEONE"
```

Related Commands None

MMEMory:LOAD:SETTings

This command restores the instrument state from a disk file into a stored setup location (buffers 1 through 5) or the current instrument state (buffer 0). Once settings have been loaded into one of the stored setup buffers (buffers 1 through 5), you can give a *RCL command to place them into buffer 0 (current instrument state).

Syntax

```
MMEMory:LOAD:SETTings <file name>, <buffer>
```

Parameters

<file name> (string)	description
A maximum of eight characters in the form “file name”	File name you want to load the setups from
<buffer> (NR1-numeric)	**description**
Valid values are 0 through 5	Buffer 0 contains the current setups; buffers 1 through 5 contain the previous setups

Dependencies A disk must be in the disk drive.

Errors and Events 402, “Operation complete; Recall of instrument setup complete”
252, “Missing media; Disk not present in drive”

Examples

```
MMEMORY:LOAD:SETTINGS "TEMP0001",0
```

Related Commands *RCL

MMEMory:STORe:TEST

This command stores the current instrument test to a disk file.

Syntax

```
MMEMory:STORe:TEST <file name>
```

Parameters

<file name> (string)	description
A maximum of eight characters in the form "file name"	File name you want to save the test to; the extension ".TST" is added by the software

Dependencies A disk must be in the disk drive.

Errors and Events 252, "Missing media; Disk not present in drive"
254, "Media full; Disk is full"
257, "File name error; File name required"
258, "Media protected; Disk is write-protected, cannot write to file"

Examples

```
MMEMORY:STORe:TEST "TEMP0001"
```

Related Commands *SAV

MMEMory:DELete:TEST

This command deletes a test file from the disk.

Syntax

```
MMEMory:DELete:TEST <file name>
```

Parameters

<file name> (string)	description
A maximum of eight characters in the form "file name"	Name of the test file you want deleted

Dependencies A disk must be in the disk drive.

Errors and Events 252, "Missing media; Disk not present in drive"
256, "File name not found; Could not open disk file"
258, "Media protected; Disk is write-protected, cannot write to file"

Examples

```
MMEMORY:DELETE:TEST "FILEONE"
```

Related Commands None

MMEMory:LOAD:TEST

This command restores a test from a disk file into the current instrument test (buffer 0).

Syntax `MMEMory:LOAD:TEST <file name>`

Parameters

<file name> (string)	description
A maximum of eight characters in the form "file name"	File name you want to load the test from

Dependencies Disk must be in the disk drive.

Errors and Events 252, "Missing media; Disk not present in drive"

Examples `MMEMORY:LOAD:TEST "TEMP0001"`

Related Commands *RCL

MMEMory:STORe:RESults

This command stores the current or stored test results to a disk file. If the current test results are of great length, only buffer number 1 may be available to store to disk. Use the SENSe:DATA:TELecom:MEASurement:BUFFer:AVAIlable? query to determine if buffer 2 exists.

Syntax `MMEMory:STORe:RESults <file name>,<buffer>`

Parameters

<file name> (string)	description
A maximum of eight characters in the form "file name"	File name you want to save the test results to; the extension ".RES" is added by the software
<buffer> (NR1-numeric)	**description**
Valid values are 1 and 2	Buffer 1 contains the current test results; buffer 2 contains the previous test results

Dependencies A disk must be in the disk drive.

Errors and Events 252, “Missing media; Disk not present in drive”
254, “Media full; Disk is full”
257, “File name error; File name required”
258, “Media protected; Disk is write-protected, cannot write to file”

Examples

```
MMEMORY:STORE:RESULTS "TEMP0001",1
```

Related Commands *SAV

MMEMory:DELete:RESults

This command deletes a test results file from the disk.

Syntax

```
MMEMory:DELete:RESults <file name>
```

Parameters

<file name> (string)	description
A maximum of eight characters in the form “file name”	Name of the test results file you want deleted

Dependencies A disk must be in the disk drive.

Errors and Events 252, “Missing media; Disk not present in drive”
256, “File name not found; Could not open disk file”
258, “Media protected; Disk is write-protected, cannot write to file”

Examples

```
MMEMORY:DELETE:RESULTS "FILEONE"
```

Related Commands None

MMEMory:LOAD:RESults

This command restores the instrument state from a disk file into a read only buffer (buffer number –1).

Syntax

```
MMEMory:LOAD:RESults <file name>
```

Parameters

<file name> (string)	description
A maximum of eight characters in the form “file name”	File name you want to load the test results from

Dependencies

Disk must be in the disk drive.

Errors and Events

252, “Missing media; Disk not present in drive”

Examples

```
MMEMORY:LOAD:RESULTS "TEMP0001"
```

Related Commands

*RCL

DISPlay Subsystem

This section describes each of the commands and queries that control aspects of the front panel display. Figure 2–45 shows the hierarchy tree for this subsystem.

Figure 2–45: DISPlay Subsystem

DISPlay:BRIGhtness

This command controls the display brightness.

Syntax `DISPlay:BRIGhtness <display brightness>`

Parameters

<display brightness> (NR3-numeric)	description
Any number between 0 and 1	0 indicates minimum brightness, 0.5 indicates medium brightness, and 1 indicates maximum brightness (default = 1.00)

Dependencies None

Errors and Events None

Examples `DISPLAY:BRIGHTNESS .8`

Related Commands None

DISPlay:BRIGhtness?

This query returns display brightness setting.

Syntax `DISPlay:BRIGhtness?`

Response

<display brightness> (NR3-numeric)	description
Any number between 0 and 1	0 indicates minimum brightness, 0.5 indicates medium brightness, and 1 indicates maximum brightness

Dependencies None

Errors and Events None

Examples Query: `DISPLAY:BRIGHTNESS?`

Response: `8.00E-1`

Related Commands DISPlay:BRIGhtness

HCOPy Subsystem

This section describes the commands and queries that control the printer. Figure 2–46 shows the hierarchy tree for this subsystem.

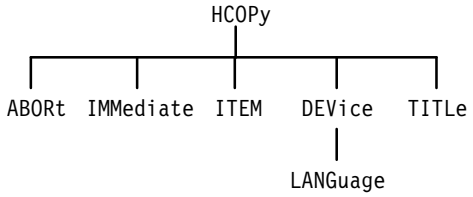


Figure 2–46: HCOPy Subsystem

HCOPy:ABORt

This command stops the current printout.

Syntax `HCOPy:ABORt`

Parameters None

Dependencies None

Errors and Events None

Examples `HCOPY:ABORT`

Related Commands HCOPy:IMMediate

HCOPy:IMMediate

This command starts the printout. The type of report to be printed is controlled with the HCOPY:ITEM command. The OPC bit will be set when the report has printed.

Syntax `HCOPy:IMMediate`

Parameters None

Dependencies Your printer type must match the HCOPy:DEVice:LANGuage setting

Errors and Events 402, “Operation complete; Hardcopy complete”
200, “Execution error; Hardcopy already in progress”

Examples

```
HCOPY:IMMEDIATE
```

Related Commands HCOPy:ABORt

HCOPy:ITEM

This command specifies the information to be printed.

Syntax

```
HCOPy:ITEM <hcopy item>
```

Parameters

<hcopy item> (discrete)	description
SUMMary	Summary of test results is printed (default)
ALLRESULTS	Summary of all measurement test results is printed
SCREEN	The current screen is printed

Dependencies None

Errors and Events None

Examples

```
HCOPY:ITEM SUMMARY
```

Related Commands HCOPy:IMMediate

HCOPy:ITEM?

This query returns which information is to be printed.

Syntax `HCOPy:ITEM?`

Response

<hcopy item> (discrete)	description
SUMMary	Summary of test results is printed (default)
ALLRESULTS	Summary of all measurement test results is printed
SCREEN	The current screen is printed

Dependencies None

Errors and Events None

Examples Query: `HCOPY:ITEM?`

Response: `ALLRESULTS`

Related Commands HCOPy:ITEM

HCOPy:DEVice:LANGuage

This command specifies the printer language.

Syntax `HCOPy:DEVice:LANGuage <printer language>`

Parameters

<printer language> (discrete)	description
TEK	Tektronix DPU-411 printer (default)
EPSOn	Epson printer
PCL	Printer Control Language (Hewlett-Packard Thinkjet)

Dependencies None

Errors and Events None

Examples `HCOPY:DEVICE:LANGUAGE EPSON`

Related Commands None

HCOPy:DEVice:LANGuage?

This query returns the printer language.

Syntax `HCOPy:DEVice:LANGuage?`

Response

<printer language> (discrete)	description
TEK	Tektronix DPU-411 printer (default)
EPSOn	Epson printer
PCL	Printer Control Language (Hewlett-Packard Thinkjet)

Dependencies None

Errors and Events None

Examples Query: `HCOPY:DEVICE:LANGUAGE?`

Response: `EPSON`

Related Commands HCOPy:DEVice:LANGuage

HCOPy:TITLe

This command enables or disables a title to be printed. If the HCOPy:TITLe is set to ON, the system owner and system user values will be printed as the title.

Syntax `HCOPy:TITLe <title enabled>`

Parameters

<title enabled> (boolean)	description
0 or OFF	Title not printed (default)
1 or ON	Title printed

Dependencies None

Errors and Events None

Examples

```
HCOPY:TITLE ON
```

Related Commands None

HCOPy:TITLe?

This query returns the current setting of the title enable. If the HCOPy:TITLe is set to 1 (ON), the system owner and system user values will be printed as the title.

Syntax

```
HCOPy:TITLe?
```

Response

<title enabled> (boolean)	description
0	Title not printed (default)
1	Title printed

Dependencies None

Errors and Events None

Examples

Query:

```
HCOPY:TITLE?
```

Response: 1

Related Commands None

SYSTem:COMMunicate:SERial Subsystem

This section describes the commands and queries that control communication over the serial port. Figure 2–47 shows the hierarchy tree for this subsystem.

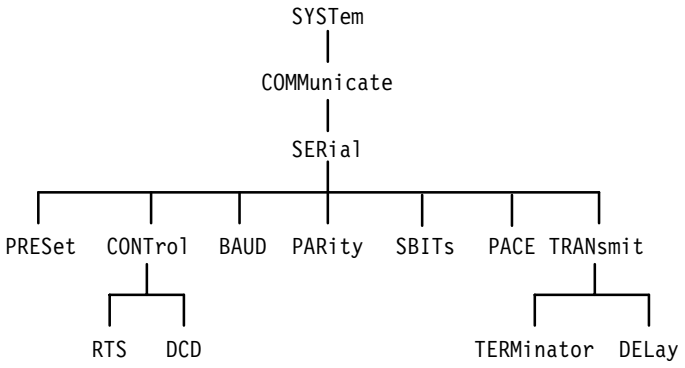


Figure 2–47: SYSTem:COMMunicate:SERial Subsystem

SYSTem:COMMunicate:SERial:PRESet

Preset sets all parameters in the serial subsystem to default values.

Syntax `SYSTem:COMMunicate:SERial:PRESet`

Parameters There are no parameters for this command. The resulting defaults for each command are shown in Table 2–36.

Table 2–36: Commands and Default Values Set by the SYSTem:COMMunicate: SERial:PRESet Command

Command	Default value
SYSTem:COMMunicate:SERial:CONtrol:DCD	0
SYSTem:COMMunicate:SERial:CONTrol:RTS	RFR
SYSTem:COMMunicate:SERial:PACE	NONE
SYSTem:COMMunicate:SERial:BAUD	9600
SYSTem:COMMunicate:SERial:PARITY	NONE
SYSTem:COMMunicate:SERial:SBITs	1
SYSTem:COMMunicate:SERial:TRANsmit:TERMinator	LF
SYSTem:COMMunicate:SERial:TRANsmit:DELay	0

Dependencies None

Errors and Events None

Examples SYSTEM:COMMUNICATE:SERIAL:PRESET

Related Commands None

SYSTem:COMMunicate:SERial:CONTrol:DCD

This command sets Data Carrier Detect (DCD) sensitivity on and off.

Syntax SYSTem:COMMunicate:SERial:CONTrol:DCD <DCD mode>

Parameters

<DCD mode> (boolean)	description
1 or ON	DCD on (default)
0 or OFF	DCD off

Dependencies None

Errors and Events None

Examples SYSTEM:COMMUNICATE:SERIAL:CONTROL:DCD OFF

Related Commands None

SYSTem:COMMunicate:SERial:CONTrol:DCD?

This query returns the current setting of Data Carrier Detect (DCD) sensitivity.

Syntax SYSTem:COMMunicate:SERial:CONTrol:DCD?

Response

<DCD mode> (boolean)	description
1	DCD on (default)
0	DCD off

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:COMMUNICATE:SERIAL:CONTROL:DCD?`

Response: 0

Related Commands SYSTem:COMMunicate:SERial:CONTrol:DCD

SYSTem:COMMunicate:SERial:CONTrol:RTS

This command sets the hardware handshaking scheme. When set to ON, handshaking is off (the instrument is not sensitive to the CTS 710). When set to RFR or IBFULL, the normal RTS/CTS hardware handshaking is on (the instrument indicates ready to receive with the RTS line).

Syntax `SYSTem:COMMunicate:SERial:CONTrol:RTS <serial RTS>`

Parameters

<serial RTS> (discrete)	description
ON	Not sensitive to the CTS 710; RTS line always asserted (handshaking is off)
RFR	Ready For Receiving (handshaking is on)
IBFULL	same as RFR

Dependencies None

Errors and Events None

Examples `SYSTEM:COMMUNICATE:SERIAL:CONTROL:RTS RFR`

Related Commands None

SYSTem:COMMunicate:SERial:CONTrol:RTS?

This query returns the current setting of the hardware handshaking scheme.

Syntax `SYSTem:COMMunicate:SERial:CONTrol:RTS?`

Response

<serial RTS> (discrete)	description
ON	Not sensitive to the CTS 710; RTS line always asserted (handshaking is off)
RFR	Ready For Receiving (handshaking is on)
IBFULL	Same as RFR

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:COMMUNICATE:SERIAL:CONTROL:RTS?`

Response: `RFR`

Related Commands SYSTem:COMMunicate:SERial:CONTrol:RTS

SYSTem:COMMunicate:SERial:BAUD

This command sets the baud rate (both transmit and receive) of the serial port.

Syntax `SYSTem:COMMunicate:SERial:BAUD <serial baud>`

Parameters

<serial baud> (NR1-numeric)	description
9600	9600 baud (default)
1200	1200 baud
2400	2400 baud
4800	4800 baud

Dependencies None

Errors and Events None

Examples

```
SYSTEM:COMM:SER:BAUD 1200
```

Related Commands None

SYSTem:COMMunicate:SERial:BAUD?

This query returns the baud rate of the serial port.

Syntax

```
SYSTem:COMMunicate:SERial:BAUD?
```

Response

<serial baud> (NR1-numeric)	description
9600	9600 baud (default)
1200	1200 baud
2400	2400 baud
4800	4800 baud

Dependencies None

Errors and Events None

Examples

Query: `SYSTEM:COMMUNICATE:SERIAL:BAUD?`

Response: `2400`

Related Commands SYSTem:COMMunicate:SERial:BAUD

SYSTem:COMMunicate:SERial:PARity

This command sets the parity of the serial port.

Syntax `SYSTem:COMMunicate:SERial:PARity <serial parity>`

Parameters

<serial parity> (discrete)	description
NONE	No parity is sent or received (default)
EVEN	Even parity is sent and received
ODD	Odd parity is sent and received

Dependencies None

Errors and Events None

Examples `SYSTEM:COMMUNICATE:SERIAL:PARITY EVEN`

Related Commands None

SYSTem:COMMunicate:SERial:PARity?

This query returns the parity of the serial port.

Syntax `SYSTem:COMMunicate:SERial:PARity?`

Response

<serial parity> (discrete)	description
NONE	No parity is sent or received (default)
EVEN	Even parity is sent and received
ODD	Odd parity is sent and received

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:COMMUNICATE:SERIAL:PARITY?`

Response: `EVEN`

Related Commands SYSTem:COMMunicate:SERial:PARity

SYSTem:COMMunicate:SERial:SBITs

This command sets the number of stop bit used by the serial port.

Syntax `SYSTem:COMMunicate:SERial:SBITs <serial sbits>`

Parameters

<serial sbits> (NR1-numeric)	description
1 or 2	Number of stop bits used by the serial port

Dependencies None

Errors and Events None

Examples `SYSTEM:COMMUNICATE:SERIAL:SBITS 1`

Related Commands None

SYSTem:COMMunicate:SERial:SBITs?

This query returns the number of stop bits used by the serial port.

Syntax `SYSTem:COMMunicate:SERial:SBITs?`

Response

<serial sbits> (NR1-numeric)	description
1 or 2	Number of stop bits used by the serial port

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:COMMUNICATE:SERIAL:SBITS?`

Response: 1

Related Commands SYSTem:COMMunicate:SERial:SBITs

SYSTem:COMMunicate:SERial:PACE

This command sets the software pacing scheme.

Syntax `SYSTem:COMMunicate:SERial:PACE <serial pace>`

Parameters

<serial pace> (discrete)	description
NONE	Software pacing disabled (default)
XON	Software pacing on using control-S and control-Q

Dependencies None

Errors and Events None

Examples `SYSTEM:COMMUNICATE:SERIAL:PACE NONE`

Related Commands None

SYSTem:COMMunicate:SERial:PACE?

This query returns the current setting of the software pacing scheme.

Syntax `SYSTem:COMMunicate:SERial:PACE?`

Response

<serial pace> (discrete)	description
NONE	Software pacing disabled (default)
XON	Software pacing on using control-S and control-Q

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:COMMUNICATE:SERIAL:PACE?`

Response: `NONE`

Related Commands SYSTem:COMMunicate:SERial:PACE

SYSTem:COMMunicate:SERial:TRANsmit:DELay

This command sets the delay, in seconds, that the instrument waits after receiving a query and before issuing a response.

Syntax `SYSTem:COMMunicate:SERial:TRANsmit:DELay <transmit delay>`

Parameters

<transmit delay> (NR1-numeric)	description
Any integer in the range 0 to 60	Delay in seconds that instrument waits after receiving a query before issuing a response (default = 0)

Dependencies None

Errors and Events None

Examples

```
SYSTEM:COMMUNICATE:SERIAL:TRANSMIT:DELAY 5
```

Related Commands None

SYSTem:COMMunicate:SERial:TRANsmit:DELay?

This query returns the current setting of the transmit delay in seconds.

Syntax

```
SYSTem:COMMunicate:SERial:TRANsmit:DELay?
```

Response

<transmit delay> (NR1-numeric)	description
Any integer in the range 0 to 60	Delay in seconds that instrument waits after receiving a query before issuing a response (default = 0)

Dependencies None

Errors and Events None

Examples

Query:

```
SYSTEM:COMMUNICATE:SERIAL:TRANSMIT:DELAY?
```

Response: 5

Related Commands SYSTem:COMMunicate:SERial:TRANsmit:DELay

SYSTem:COMMunicate:SERial:TRANsmit:TERMinator

This command sets the character or characters sent with the response message.

Syntax `SYSTem:COMMunicate:SERial:TRANsmit:TERMinator <terminator>`

Parameters

<terminator> (discrete)	description
LF	Line feed (default)
CR	Carriage return
CRLF	CR/LF
LFCR	LF/CR

Dependencies None

Errors and Events None

Examples `SYSTEM:COMMUNICATE:SERIAL:TRANSMIT:TERMINATOR CRLF`

Related Commands None

SYSTem:COMMunicate:SERial:TRANsmit:TERMinator?

This query returns the current setting of the character or characters sent with the response message.

Syntax `SYSTem:COMMunicate:SERial:TRANsmit:TERMinator?`

Response

<terminator> (discrete)	description
LF	Line feed (default)
CR	Carriage return
CRLF	CR/LF
LFCR	LF/CR

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:COMMUNICATE:SERIAL:TRANSMIT:TERMINATOR?`

Response: `CRLF`

Related Commands SYSTem:COMMunicate:SERial:TRANsmit:TERMinator

Instrument Control Commands

The Instrument Control commands allow you to access utility functions and settings such as errors, query headers, serial number, system time, date, owner, and operator. This section contains all of the commands and queries for the following Instrument Control subsystem:

- SYSTem

SYSTem Subsystem

This section describes each of the commands and queries that allow you to access general utility functions and settings in the instrument. Figure 2–48 shows the hierarchy tree for this subsystem.

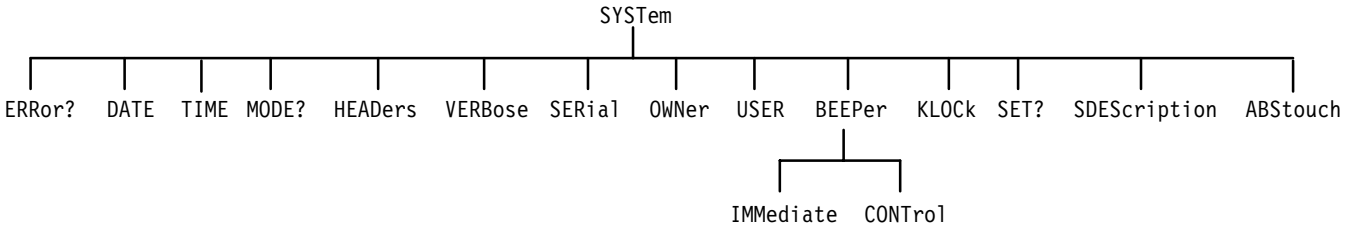


Figure 2–48: SYSTem Subsystem

SYSTem:ERRor?

This query returns the errors and events that have accumulated in the instrument. If no errors are present in the instrument, the response is: 0,“No Error”.

Syntax `SYSTem:ERRor?`

Response

<error number> (NR1-numeric)	description
Any integer in the range 0 to 999	This value indicates the error number
<error description> (string)	**description**
Primary error message and, optionally, a secondary error message	This string describes the error

Dependencies None

Errors and Events See the *Messages* tables in the *Status and Events* section.

Examples Query: `SYSTEM:ERROR?`

Response: 200, “Execution error; Pointer burst active, request ignored”

Related Commands None

SYSTem:DATE

This command sets the date for the instrument.

Syntax `SYSTem:DATE <year>,<month>,<day>`

Parameters

<year> (NR1-numeric)	description
Any integer in the range 0 to 99	The system year is set to this value
<month> (NR1-numeric)	**description**
Any integer in the range 1 to 12	The system month is set to this value
<day> (NR1-numeric)	**description**
Any integer in the range 1 to 31	The system day is set to this value

Dependencies None

Errors and Events None

Examples `SYSTEM:DATE 93,12,1`

Related Commands SYSTem:TIME

SYSTem:DATE?

This query returns the date in the instrument.

Syntax `SYSTem:DATE?`

Response

<year> (NR1-numeric)	description
Any integer in the range 0 to 99	The system year is set to this value
<month> (NR1-numeric)	**description**
Any integer in the range 1 to 12	The system month is set to this value
<day> (NR1-numeric)	**description**
Any integer in the range 1 to 31	The system day is set to this value

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:DATE?`

Response: `95,12,1`

Related Commands SYSTem:DATE

SYSTem:TIME

This command sets the time for the instrument. Time is kept in a 24-hour format.

Syntax `SYSTem:TIME <hour>,<minute>,<second>`

Parameters

<hour> (NR1-numeric)	description
Any integer in the range 0 to 23	The system hour is set to this value
<minute> (NR1-numeric)	**description**
Any integer in the range 0 to 59	The system minute is set to this value

(continued on next page)

<second> (NR1-numeric)	description
Any integer in the range 0 to 59	The system second is set to this value

Dependencies None

Errors and Events None

Examples `SYSTEM:TIME 13,7,56`

Related Commands SYSTem:DATE

SYSTem:TIME?

This query returns the time in the instrument. Time is kept in a 24-hour format.

Syntax `SYSTem:TIME?`

Response

<hour> (NR1-numeric)	description
Any integer in the range 0 to 23	The system hour is set to this value
<minute> (NR1-numeric)	**description**
Any integer in the range 0 to 59	The system minute is set to this value
<second> (NR1-numeric)	**description**
Any integer in the range 0 to 59	The system second is set to this value

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:TIME?`

Response: `10,15,3`

Related Commands SYSTem:TIME

SYSTem:MODE?

This query returns the mode of the instrument.

Syntax `SYSTem:MODE?`

Response

<system mode> (discrete)	description
SONet	Instrument is set to SONET mode

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:MODE?`

Response: `SONET`

Related Commands SYSTem:MODE

SYSTem:HEADers

This command controls the presence of headers in query responses.

Syntax `SYSTem:HEADers <system headers>`

Parameters

<system headers> (boolean)	description
0 or OFF	No system headers are returned (default)
1 or ON	System headers are returned

Dependencies None

Errors and Events None

Examples `SYSTEM:HEADERS OFF`

Related Commands None

SYSTem:HEADers?

This query returns the current setting of header control.

Syntax `SYSTem:HEADers?`

Response

<system headers> (boolean)	description
0	No system headers are returned (default)
1	System headers are returned

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:HEADERS 1;SYSTEM:HEADERS?`

Response: `SYST:HEAD 1` **or** `SYSTEM:HEADERS 1`

Related Commands SYSTem:HEADers

SYSTem:VERBose

This command controls the length of headers in query responses. If verbose is ON, the long form of headers is returned. If verbose is OFF, the short form is returned.

Syntax `SYSTem:VERBose <system verbose>`

Parameters

<system verbose> (boolean)	description
0 or OFF	Short form of headers (default)
1 or ON	Long form of headers

Dependencies SYSTem:HEADers must be set to ON for headers to be returned.

Errors and Events None

Examples

```
SYSTEM:VERBOSE OFF
```

Related Commands None

SYSTem:VERBose?

This query returns the current setting of verbose control.

Syntax

```
SYSTem:VERBose?
```

Response

<system verbose> (boolean)	description
0	Short form of headers (OFF) (default)
1	Long form of headers (ON)

Dependencies SYSTem:HEADers must be set to ON for headers to be returned.

Errors and Events None

Examples

Query:

```
SYSTEM:VERBOSE OFF;SYSTEM:VERBOSE?
```

Response:

```
SYST:VERB 0
```

Query:

```
SYSTEM:VERBOSE ON;SYSTEM:VERBOSE?
```

Response:

```
SYSTEM:VERBOSE 1
```

Related Commands SYSTem:VERBose

SYSTem:SERial

This command sets the instrument serial number. The factory assigns the serial number; however, you may alter the serial number. Only ASCII alphanumeric characters are accepted.

Syntax

```
SYSTem:SERIal <serial number>
```

Parameters

<serial number> (string)	description
Any ASCII string, maximum length of 16	The instrument serial number is set to this value

Dependencies None

Errors and Events None

Examples

```
SYSTEM:SERIAL "B010100"
```

Related Commands *IDN?

SYSTem:SERial?

This query returns the instrument serial number.

Syntax

```
SYSTem:SERIal?
```

Response

<serial number> (string)	description
Any ASCII string, maximum length of 16	The instrument serial number is set to this value

Dependencies None

Errors and Events None

Examples

Query: `SYSTEM:SERIAL?`

Response: `"B010100"`

Related Commands

SYSTem:SERIal
*IDN?

SYSTem:OWNer

This command sets the instrument owner. This information is saved in the results buffer and printed in hardcopy reports.

Syntax

`SYSTem:OWNer <system owner>`

Parameters

<system owner> (string)	description
Any ASCII string, maximum length of 64	The instrument system owner is set to this value

Dependencies

None

Errors and Events

None

Examples

`SYSTEM:OWNER "The ABC TELECOM COMPANY"`

Related Commands

None

SYSTem:OWNer?

This query returns the instrument owner.

Syntax

`SYSTem:OWNer?`

Response

<system owner> (string)	description
Any ASCII string, maximum length of 64	The instrument system owner is set to this value

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:OWNER?`

Response: `"The ABC TELECOM COMPANY"`

Related Commands SYSTem:OWNer

SYSTem:USER

This command sets the instrument operator name. This information is saved in the results buffer and printed in hardcopy reports.

Syntax `SYSTem:USER <operator name>`

Parameters

<operator name> (string)	description
Any ASCII string, maximum length of 64	The instrument operator name is set to this value

Dependencies None

Errors and Events None

Examples `SYSTEM:USER "JOHN DOE"`

Related Commands SYSTem:OWNer

SYSTem:USER?

This query returns the instrument operator name.

Syntax `SYSTem:USER?`

Response

<operator name> (string)	description
Any ASCII string, maximum length of 64	The instrument operator name is set to this value

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:USER?`

Response: `"JOHN DOE"`

Related Commands SYSTem:USER

SYSTem:BEEPer:IMMediate

This command is used to sound the beeper in the instrument.

Syntax `SYSTem:BEEPer:IMMediate`

Parameters None

Dependencies None

Errors and Events None

Examples `SYSTEM:BEEPER:IMMEDIATE`

Related Commands SYSTem:BEEPer:CONTrol

SYSTem:BEEPer:CONTrol

This command sets the beeper control.

Syntax `SYSTem:BEEPer:CONTrol <beeper control>`

Parameters

<beeper control> (boolean)	description
0 or OFF	Use SYSTem:BEEPer:IMMediate to sound the beeper (default)
1 or ON	The beeper sounds when any error, alarm, or failure is detected

Dependencies None

Errors and Events None

Examples

```
SYSTEM:BEEPER:CONTROL ON
```

Related Commands None

SYSTem:BEEPer:CONTrol?

This query returns the current setting of the beeper control.

Syntax

```
SYSTem:BEEPer:CONTrol?
```

Response

<beeper control> (boolean)	description
0	Use SYSTem:BEEPer:IMMediate to sound the beeper (default)
1	The beeper sounds when any error, alarm, or failure is detected

Dependencies None

Errors and Events None

Examples

Query: `SYSTEM:BEEPER:CONTROL?`

Response: `0`

Related Commands

SYSTem:BEEPer:CONTrol

SYSTem:KLOCk

This command enables and disables front panel control. (KLOCk stands for keyboard lock.)

Syntax

`SYSTem:KLOCk <front panel control>`

Parameters

<front panel control> (boolean)	description
0 or OFF	Front panel enabled (default)
1 or ON	Front panel disabled

Dependencies

None

Errors and Events

None

Examples

`SYSTEM:KLOCK ON`

Related Commands

None

SYSTem:KLOCk?

This query returns the current setting of the front panel control. (KLOCk stands for keyboard lock.)

Syntax

`SYSTem:KLOCk?`

Response

<front panel control> (boolean)	description
0	Front panel enabled (default)
1	Front panel disabled

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:KLOCk?`

Response: `0`

Related Commands SYSTem:KLOCk

SYSTem:SET?

This query returns the current instrument state and performs the same function as the *LRN? query.

Syntax `SYSTem:SET?`

Response A list of commands and their parameter values separated by semicolons (;) (see *Appendix C* for a complete list).

Dependencies None

Errors and Events None

Examples Query: `SYSTEM:SET?`

Response:
```
:OUTPUT1:TELECOM:RATE STS1;TYPE ELECTRICAL;LEVEL
XCONNECT;:SOURCE:CLOCK:SOURCE INTERNAL;OFFSET:MODE
LOFFSET;LVALUE 0;:SOURCE:DATA:TELECOM:SOURCE OUTPUT1;
...
```

Related Commands *LRN?

SYSTem:SDEScription

This command sets the description for the stored settings in the current buffer.

Syntax `SYSTem:SDEScription <description>`

Parameters

<description> (string)	description
Any string, maximum length of 24 characters	Description of the stored settings in the current buffer

Dependencies After you set the description with this command, use the *SAV command to save the description and settings in memory or the MMEMory:STORe:SETTings command to save the description and settings to disk.

Errors and Events None

Examples

```
SYSTEM:SDESCRIPTION "PASS/FAIL TEST A001"
```

Related Commands *SAV
*RCL
MMEMory:STORe:SETTings

SYSTem:SDEScription?

This query returns the description for the stored settings in the current buffer.

Syntax

```
SYSTem:SDEScription?
```

Response

<description> (string)	description
Any string, maximum length of 24 characters	Description of the stored strings in the current buffer

Dependencies None

Errors and Events None

Examples Query:

```
SYSTEM:SDESCRIPTION?
```

Response:

```
"PASS/FAIL TEST A001"
```

Related Commands SYSTem:SDEScription

SYSTem:ABStouch

This command simulates front panel button presses and knob turns.

Syntax `SYSTem:ABStouch <abstouch>`

Parameters

<abstouch> (discrete)	description
HB1	Horizontal bezel #1 (left-most button)
HB2	Horizontal bezel #2
HB3	Horizontal bezel #3
HB4	Horizontal bezel #4
HB5	Horizontal bezel #5
HB6	Horizontal bezel #6
HB7	Horizontal bezel #7 (right-most button)
VB1	Vertical bezel #1 (top button)
VB2	Vertical bezel #2
VB3	Vertical bezel #3
VB4	Vertical bezel #4
VB5	Vertical bezel #5 (bottom button)
STESTs	Stored tests menu
TRANsmit	TRANSMIT menu
RECEiver	RECEIVE menu
RESults	RESULTS menu
UTILity	UTILITY menu
STARTSTOP	START/STOP button
STOP	START/STOP button
IERRor	INSERT ERROR button
POINter	POINTER ACTION button
RHIStory	CLEAR HISTORY button
INCKnob	Increment knob
DECKnob	Decrement knob
HELP	HELP button
AUTOscan	AUTOSCAN button
PRINT	PRINT button

Dependencies None

Errors and Events None

Examples

```
SYSTEM:ABSTOUCH VB1
```

Related Commands None

Diagnostic Commands

The Diagnostic commands allow control of the diagnostic self tests provided with your instrument. This section contains all of the commands and queries for the following Diagnostic subsystem:

- DIAGnostic

DIAGnostic Subsystem

This section describes each of the commands and queries that allow access and control of the diagnostic self tests provided with your instrument. Figure 2–49 shows the hierarchy tree for this subsystem.

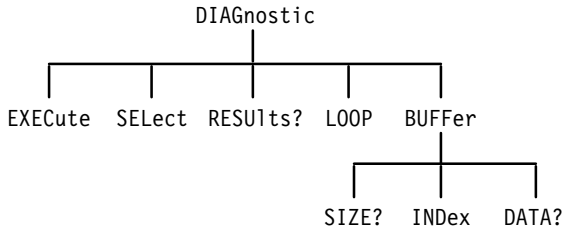


Figure 2–49: DIAGnostic Subsystem

DIAGnostic:EXECute

This command executes the selected diagnostic routines and sets the OPC bit when completed.

Syntax `DIAGnostic:EXECute`

Parameters None

Dependencies The routines executed are determined by the DIAGnostic:SELect command.

Errors and Events 200, “Execution error; Diagnostics invalid while disk or autoscan busy”
402, “Operation complete; Internal diagnostics completed – passed”
402, “Operation complete; Internal diagnostics completed – failed”

Examples

```
DIAGNOSTIC:EXECUTE
```

Related Commands

DIAGnostic:SELect
*TST?

DIAGnostic:SELect

This command reports the specified diagnostic routine that are run when the DIAGnostic:EXECute command is sent.

Syntax

```
DIAGnostic:SELect <diagnostic group>
```

Parameters

<diagnostic group> (discrete)	description
STANdard	Standard self test; same as the *TST? query (default)
PROCessor	Processor board
PROTocol	Protocol board
CLOCK	Clock generator board
DISK	Disk drive
DISPLAY	Display board
TRIButary	Tributary board (Add/Drop/Test Option Only)
INTERFACE	Line interface module
FPANEL	Front Panel
SYSINTERNAL	Complete system (internal loopback)
SYSEXTERNAL	Complete system (external loopback)

Dependencies

None

Errors and Events

None

Examples

```
DIAGNOSTIC:SELECT STANDARD
```

Related Commands

DIAGnostic:EXECute

DIAGnostic:SELect?

This query selects the current setting of the diagnostic routine that are run when the DIAGnostic:EXECute command is sent.

Syntax `DIAGnostic:SELect?`

Response

<diagnostic group> (discrete)	description
STANdard	Standard self test (default)
PROCessor	Processor board
PROTocol	Protocol board
CLOCK	Clock generator board
DISK	Disk drive
DISPLAY	Display board
TRIButary	Tributary board (Add/Drop/Test Option Only)
INTERFACE	Line interface module
FPANEL	Front Panel
SYSINTERNAL	Complete system (internal loopback)
SYSEXTERNAL	Complete system (external loopback)

Dependencies None

Errors and Events None

Examples Query: `DIAGNOSTIC:SELECT?`

Response: `STANDARD`

Related Commands DIAGnostic:SELect

DIAGnostic:RESults?

This query returns the results from the last diagnostics execution. These diagnostics could have been run at the power-on self test or as a result of sending the DIAGnostic:EXECute command or *TST? query.

If any SCPI-derived or IEEE 488.2 Common Command is sent while the diagnostics are running, the diagnostics will stop. The DIAGnostic:RESults? query can then be used to determine if the selected diagnostic test passed or failed.

Syntax

```
DIAGnostic:RESults?
```

Response

<diagnostic results> (discrete)	description
PASSED	Test passed
FAILED	Test failed

Dependencies

None

Errors and Events

None

Examples

Query: `DIAGNOSTIC:RESULTS?`

Response: `PASSED`

Related Commands

DIAGnostic:SELect
DIAGnostic:EXECute

DIAGnostic:LOOP

This command specifies the looping control used for diagnostic routines when the DIAGnostics:EXECute command is given.

Syntax

```
DIAGnostic:LOOP <loop control>
```

Parameters

<loop control> (discrete)	description
ONCE	One pass (default)
TEN	Loop ten times
THOUSAND	Loop one thousand times
ERROR	Loop until error detected
FOREver	Loop until any command is sent

Dependencies None

Errors and Events None

Examples

```
DIAGNOSTIC:LOOP TEN
```

Related Commands DIAGnostic:EXECute

DIAGnostic:LOOP?

This query returns the current setting of the looping control used for diagnostic routines when the DIAGnostics:EXECute command is given.

Syntax

```
DIAGnostic:LOOP?
```

Response

<loop control> (discrete)	description
ONCE	One pass (default)
TEN	Loop ten times
THOUSAND	Loop one thousand times
ERROR	Loop until error detected
FOREver	Loop until any command is sent

Dependencies None

Errors and Events None

Examples

Query: `DIAGNOSTIC:LOOP?`

Response: `ONCE`

Related Commands DIAGnostic:LOOP

DIAGnostic:BUFFer:SIZE?

This query returns the number of entries in the diagnostics results buffer. The <buffer size> is the maximum value you can use in the DIAGnostic:BUFFer:INDex command.

Syntax `DIAGnostic:BUFFer:SIZE?`

Response

<buffer size> (NR1-numeric)	description
Any integer	The number of entries in the results buffer (0 indicates that no errors were found in the diagnostics)

Dependencies None

Errors and Events None

Examples

Query: `DIAGNOSTIC:BUFFER:SIZE?`

Response: `5`

Related Commands DIAGnostic:BUFFer:INDex

DIAGnostic:BUFFer:INDex

This command selects the buffer that is used by the DIAGnostic:BUFFer:DATA? query. The <buffer number> must be less than or equal to the <buffer size> value returned from the DIAGnostic:BUFFer:SIZE? query. When the DIAGnostic:EXECute command is given, the <buffer number> is reset to 1.

Syntax `DIAGnostic:BUFFer:INDex <buffer number>`

Parameters

<buffer number> (NR1-numeric)	description
Any integer; must be less than or equal to the integer returned from the DIAGnostic:BUFFer:SIZE? query	Buffer used by the DIAGnostic:BUFFer:DATA? query (default = 1)

Dependencies None

Errors and Events None

Examples

```
DIAGNOSTIC:BUFFER:INDEX 1
```

Related Commands DIAGnostic:BUFFer:SIZE?
DIAGnostic:BUFFer:DATA?
DIAGnostic:BUFFer:EXECute

DIAGnostic:BUFFer:INDex?

This query returns the current setting of the buffer that is used by the DIAGnostic:BUFFer:DATA? query. When the DIAGnostic:EXECute command is given, the <buffer number> is reset to 1.

Syntax

```
DIAGnostic:BUFFer:INDex?
```

Response

<buffer number> (NR1-numeric)	description
Any integer	This buffer number is used by the DIAGnostic:BUFFer:DATA? query (default = 1)

Dependencies None

Errors and Events None

Examples

Query: `DIAGNOSTIC:BUFFER:INDEX?`

Response: 1

Related Commands

DIAGnostic:BUFFer:INDex

DIAGnostic:BUFFer:DATA?

This query returns detailed description of the diagnostic results. Use the DIAGnostic:BUFFer:INDex command to select the buffer to view.

Syntax

`DIAGnostic:BUFFer:DATA?`

Response

<diagnostic description> (string)	description
An ASCII string, maximum length 160	A detailed description of the diagnostic results

Dependencies

None

Errors and Events

None

Examples

Query: `DIAGNOSTIC:BUFFER:DATA?`

Response: `"MEMORY TEST FAILED - WROTE AA READ 55"`

Related Commands

DIAGnostic:BUFFer:INDex
DIAGnostic:SELect
DIAGnostic:EXECute

Common Commands

This section describes each of the IEEE 488.2 Common Commands in detail. See page 2–9 for information on reading the <decimal value> tables contained in this section.

*CLS

This command clears all status registers and error queues.

Syntax `*CLS`

Parameters None

Dependencies None

Errors and Events None

Examples `*CLS`

Related Commands None

*ESE

This command sets the contents of the Event Status Enable Register. This register controls the reporting of specific errors through the status register and the interrupt mechanism.

Syntax `*ESE <decimal value>`

Parameters

<decimal value> (NR1-numeric)	bit	definition
1	0	Operation complete
2	1	Not used
4	2	Query error

(continued on next page)

<decimal value> (NR1-numeric)	bit	definition
8	3	Device dependent error
16	4	Execution error
32	5	Command error
64	6	Not used
128	7	Power on

Dependencies None

Errors and Events None

Examples

```
*ESE 16
```

Related Commands *ESE?

*ESE?

This query returns the contents of the Event Status Enable Register.

Syntax

```
*ESE?
```

Response

<decimal value> (NR1-numeric)	bit	definition
1	0	Operation complete
2	1	Not used
4	2	Query error
8	3	Device dependent error
16	4	Execution error
32	5	Command error
64	6	Not used
128	7	Power on

Dependencies None

Errors and Events None

Examples Query: `*ESE?`

Response: `64`

Related Commands *ESE

*ESR?

This query returns the contents of the Standard Event Status Register. This register shows the status of general instrument-related events as bits encoded into a number.

Syntax `*ESR?`

Response

<decimal value> (NR1-numeric)	bit	definition
1	0	Operation complete
2	1	Not used
4	2	Query error
8	3	Device dependent error
16	4	Execution error
32	5	Command error
64	6	Not used
128	7	Power on

Dependencies None

Errors and Events None

Examples

Query: `*ESR?`

Response: `64`

Related Commands *ESE

*IDN?

This query returns the identity of the instrument.

Syntax `*IDN?`

Response `<manufacturer>,<model>,<serial number>,<firmware version>`

Dependencies None

Errors and Events None

Examples

Query: `*IDN?`

Response: `"TEKTRONIX,CTS710,000000000,CF:91.1C FV1.20"`

Related Commands None

*LRN?

This query returns an ASCII representation of the current instrument setup.

Syntax `*LRN?`

Response A list of commands and their parameter values separated by semicolons (;) (see *Appendix C* for a complete list).

Dependencies None

Errors and Events None

Examples

Query: `*LRN?`

Response:
```
":OUTPUT1:TELECOM:RATE STS1;TYPE ELECTRICAL;LEVEL
XCONNECT;:SOURCE:CLOCK:SOURCE INTERNAL;OFFSET:MODE
LOFFSET;LVALUE 0;:SOURCE:DATA:TELECOM:SOURCE OUTPUT1;
..."
```

Related Commands None

*RST

This command resets the instrument and puts it into a default state, which is independent of past historical setups. *Appendix D* summarizes the effect of *RST on the instrument. This command sets the Operation Complete bit in the Standard Event Status Register.

Syntax `*RST`

Parameters None

Dependencies None

Errors and Events 402, “Operation complete; Instrument factory reset complete”

Examples `*RST`

Related Commands None

*SRE

This command sets the contents of the Service Request Enable Register. This register controls the reporting of specific errors through the status register and the interrupt mechanism.

Syntax `*SRE <decimal value>`

Parameters

<decimal value> (NR1-numeric)	bit	definition
1	0	Not used
2	1	Not used
4	2	Not used
8	3	Not used
16	4	Message available
32	5	Event status summary
64	6	Not used
128	7	Not used

Dependencies None

Errors and Events None

Examples

```
*SRE 64
```

Related Commands *SRE?

*SRE?

This query returns the contents of the Service Request Enable Register.

Syntax

```
*SRE?
```

Response

<decimal value> (NR1-numeric)	bit	definition
1	0	Not used
2	1	Not used
4	2	Not used
8	3	Not used
16	4	Message available
32	5	Event status summary

(continued on next page)

<decimal value> (NR1-numeric)	bit	definition
64	6	Not used
128	7	Not used

Dependencies None

Errors and Events None

Examples Query: `*SRE?`

Response: `64`

Related Commands *SRE

*STB?

This query returns the contents of the Status Byte Register.

Syntax `*STB?`

Response

<decimal value> (NR1-numeric)	bit	definition
1	0	Not used
2	1	Not used
4	2	Error/event queue not empty
8	3	Not used
16	4	Message available
32	5	Event status summary
64	6	Request service/Master status summary
128	7	Not used

Dependencies None

Errors and Events None

Examples Query: `*STB?`

Response: `64`

Related Commands *SRE
*SRE?

*OPC

This command causes the instrument to generate the Operation Complete message in the Standard Event Status Register when all pending instrument operations have been finished.

Syntax `*OPC`

Parameters None

Dependencies None

Errors and Events None

Examples `*OPC`

Related Commands *OPC?

*OPC?

This query returns a 1 when all pending device operations have finished.

Syntax `*OPC?`

Response 1

Dependencies None

Errors and Events None

Examples Query: `*OPC?`

Response: 1

Related Commands None

*OPT?

This query returns the installed hardware options and the instrument and option configuration information. <option name> is repeated for each option; <instrument configuration> and <option configuration> are listed only once.

Syntax `*OPT?`

Response `<option name>,<instrument configuration>,<option configuration>`

Dependencies None

Errors and Events None

Examples Query: `*OPT?`

Response: `"OPT/ELEC 1310nm: 55/155/622,DS1/DS3: ADD/DROP/TEST, [C18:P0:H0:K0:D0:F8:X2.93],[I13:T3:TX1.20:J-]"`

Related Commands None

*RCL

This command recalls the instrument state from an internal storage buffer. Five buffers are available for use. This command sets the Operation Complete bit in the Standard Event Status Register.

Syntax

```
*RCL <buffer number>
```

Parameters

<buffer number> (NR1-numeric)	description
1 to 5	Storage buffers 1 to 5

Dependencies

None

Errors and Events

230, “Data corrupt or stale; Recall buffer is empty”
221, “Settings conflict; Internal buffer is empty”

Examples

```
*RCL 1
```

Related Commands

*SAV

*SAV

This command saves the instrument state into an internal storage buffer. Five buffers are available for use. This command sets the Operation Complete bit in the Standard Event Status Register.

Syntax

```
*SAV <buffer number>
```

Parameters

<buffer number> (NR1-numeric)	description
1 to 5	Storage buffers 1 to 5

Dependencies

None

Errors and Events

None

Examples *SAV 1

Related Commands *RCL

*TST?

This query invokes the instrument self-test routines and returns the result when they complete. The OPC bit in the Standard Event Status Register is set when the self-test routines are complete.

Syntax *TST?

Response

<test results> (NR1-numeric)	description
0	Test completed and was successful
1	Test completed and failed

Dependencies None

Errors and Events 402, “Operation complete; Internal diagnostics completed – passed”
402, “Operation complete; Internal diagnostics completed – failed”

Examples Query: *TST?

Response: 0

Related Commands DIAGnostics:RESults?
DIAGnostics:EXECute

*WAI

This command prevents any commands or queries from executing until the command that is currently executing sets the OPC bit.

Syntax *WAI

Parameters None

Dependencies None

Errors and Events None

Examples

```
*WAI
```

Related Commands None

Status and Events

The Status and Event Reporting System reports asynchronous events and errors that occur in the CTS 710 SONET Test Set. This system consists of four 8-bit registers and two queues that you access through the command language. You can use these registers and queues to query the instrument status and control the interrupts that report events.

In general, after an interrupt occurs, first conduct a serial poll, query the registers to see why the interrupt occurred, and then send the SYSTem:ERROr? query to see a descriptive error message.

This section describes the four registers and two queues of the Status and Event Reporting System. For each register, you are given a description, a table describing all of the bits, and an example of how to use the register. Also described in this section is the Status and Event Reporting process, synchronizing programming commands, and the system messages.

Status and Event Reporting System

The Status and Event Reporting System monitors and reports such events as an error occurring or the availability of a response to a query. This system includes descriptions of the following registers and queues:

- Status Byte Register
- Service Request Enable Register
- Standard Event Status Register
- Event Status Enable Register
- Output Queue
- System Error and Event Queue

Status Byte Register

The Status Byte Register, shown in Table 3–1, summarizes information from other registers. Use a serial poll or a *STB? query to read the contents of the Status Byte Register. The response is the sum of the decimal values for all bits set. When you use a serial poll, bit 6 shows Request Service information. When you use the *STB? query, bit 6, the Master Status Summary bit, indicates that bits 4 or 5 may be set. Using the *STB? query clears all bits in the Status Byte Register.

Table 3–1: The Status Byte Register

Bit	Decimal Value	Function
0–1	–	Not used
2	4	**Error/Event Queue not empty** indicates that information is contained in the error/event queue and is waiting to be read.
3	–	Not used
4	16	**Message Available** shows that output is available in the Output Queue.
5	32	**Event Status Bit** indicates that one or more events have occurred and the corresponding bits in the Standard Event Status Register have been set.
6	64	**Request Service** (obtained from a serial poll) shows that the CTS 710 has requested service from the GPIB controller. **Master Status Summary** (obtained from *STB? query) summarizes the event status bit and message available bits in the Status Byte Register.
7	–	Not used

A common example of using the Status Byte Register is to enable only the Event Status and Request Service bits. Enable bits 5 and 6 using the Service Request Enable Register (see the next section for information about this register). If the *STB? query returns a value of 96, bit 5 (decimal value of 32) and bit 6 (decimal value of 64) have been set (giving a decimal value sum of 96). Bit 5 indicates that information is available in the Standard Event Status Register, and bit 6 indicates that bits 4 or 5 are set in the Status Byte Register.

Service Request Enable Register

The Service Request Enable Register, shown in Table 3–2, controls which bits in the Status Byte Register will generate a service request. Use the *SRE command to set bits in the Service Request Enable Register. Use the *SRE? query to see which bits in this register are enabled. The response from this query is the sum of the decimal values for all bits set.

Table 3–2: The Service Request Enable Register

Bit	Decimal Value	Function
0–3	–	Not used
4	16	**Message Available** indicates that a message available will generate a service request.

Table 3–2: The Service Request Enable Register (Cont.)

Bit	Decimal Value	Function
5	32	**Event Status Bit** indicates that events summarized in bit 5 of the Status Byte Register will generate a service request.
6–7	–	Not used

If, for example, the *SRE? query returns a value of 48, bits 4 and 5 are set in the Service Request Enable Register. Any event that causes the Message Available bit (bit 4) or Event Status bit (bit 5) to be set in the Status Byte Register now generates an interrupt. If you want an interrupt to be generated only when the Event Status bit (bit 5) is set, use the *SRE 32 command.

Standard Event Status Register

The Standard Event Status Register, shown in Table 3–3, records many types of events that can occur in the CTS 710. Use the *ESR? query to read the contents of this register. The response is the sum of the decimal values for all bits set. Reading this register clears all bits so the register can accumulate information about new events.

Table 3–3: The Standard Event Status Register

Bit	Decimal Value	Function
0	1	**Operation Complete** shows that the operation is complete. This bit is set when all pending operations complete following a *OPC command. Table B–1 in the Appendix lists the commands and queries that set the OPC bit upon completion of execution.
1	–	Not used
2	4	**Query Error** shows that the CTS 710 attempted to read the Output Queue when no data was present or pending, or that data in the Output Queue was lost.
3	8	**Device Dependent Error** shows that a device error occurred. Table 3–7 on page 3–10 lists the device error messages.
4	16	**Execution Error** shows that an error occurred while the CTS 710 was executing a command or query. Table 3–6 on page 3–9 lists the execution error messages.
5	32	**Command Error** shows that an error occurred while the CTS 710 was parsing a command or query. Table 3–5 on page 3–8 lists the command error messages.

Table 3–3: The Standard Event Status Register (Cont.)

Bit	Decimal Value	Function
6	–	Not used
7	128	**Power On** shows that the CTS 710 was powered on. The completion of the diagnostic tests also sets this bit.

The following example assumes that all bits have been enabled using the Event Status Enable Register (see the next section for information about this register). If a *ESR? query returns a value of 128, bit 7 (decimal value of 128) is set indicating that the instrument is in the initial power-on state.

Table B–1 in the Appendix lists the commands and queries that set the OPC bit (bit 0 of the Standard Event Status Register) upon completion of execution. Some of these commands and queries may require more than 200 ms to complete execution.

Event Status Enable Register

The Event Status Enable Register, shown in Table 3–4, controls which events are summarized in the event status bit (bit 5) of the Status Byte Register. Note that the Event Status Enable Register has the same content as the Standard Event Status Register. Use the *ESE command to set bits in the Event Status Enable Register. Use the *ESE? query to see what bits in the Event Status Enable Register are set. The response from this query is the sum of the decimal values for all bits summarized in the event status bit of the Status Byte Register.

Table 3–4: The Event Status Enable Register

Bit	Decimal Value	Function
0	1	**Operation Complete** shows that the operation is complete. This bit is set when all pending operations complete following a *OPC command. Table B–1 in the Appendix lists the commands and queries that set the OPC bit upon completion of execution.
1	–	Not used
2	4	**Query Error** shows that the CTS 710 attempted to read the Output Queue when no data was present or pending, or that data in the Output Queue was lost.
3	8	**Device Dependent Error** shows that a device error occurred. Table 3–7 on page 3–10 lists the device error messages.
4	16	**Execution Error** shows that an error occurred while the CTS 710 was executing a command or query. Table 3–6 on page 3–9 lists the execution error messages.

Table 3–4: The Event Status Enable Register (Cont.)

Bit	Decimal Value	Function
5	32	**Command Error** shows that an error occurred while the CTS 710 was parsing a command or query. Table 3–5 on page 3–8 lists the command error messages.
6	–	Not used
7	128	**Power On** shows that the CTS 710 was powered on. The completion of the diagnostic tests also sets this bit.

If, for example, the *ESE? query returns a value of 255, all bits are set indicating that all events will set the event status bit (bit 5) of the Status Byte Register.

The Output Queue

The CTS 710 stores query responses in the Output Queue. It empties this queue each time it receives a new command or query message after an End Of Message (EOM). The controller must read a query response before it sends the next command (or query) or it loses responses to earlier queries.

***NOTE**. When a controller sends a query, an EOM, and a second query, the CTS 710 normally clears the first response and outputs the second while reporting a Query Error (bit 2 in the Standard Event Status Register) to indicate the lost response.*

The System Error and Event Queue

The CTS 710 error and event messages are stored in the System Error and Event Queue. Use the SYSTem:ERRor? query to get the event number and a text description of the event. Reading an event removes it from the queue. The Event Queue stores detailed information for up to 20 events; the events are stored in first-in first-out order.

Before reading an event from the Event Queue, use the *ESR? query to read the summary of the event from the Standard Event Status Register. The events summarized by the *ESR? query are made available to the SYSTem:ERRor? query, and the Standard Event Status Register is emptied.

Reading the Standard Event Status Register erases any events that were summarized by previous *ESR? queries but not read from the Event Queue. Events that follow an *ESR? query are put in the Event Queue but are not available until *ESR? is given again.

Status and Event Reporting Process

Figure 3–1 shows how to use the Status and Event Reporting system. In the explanation that follows, numbers in parentheses refer to the circled numbers in Figure 3–1.

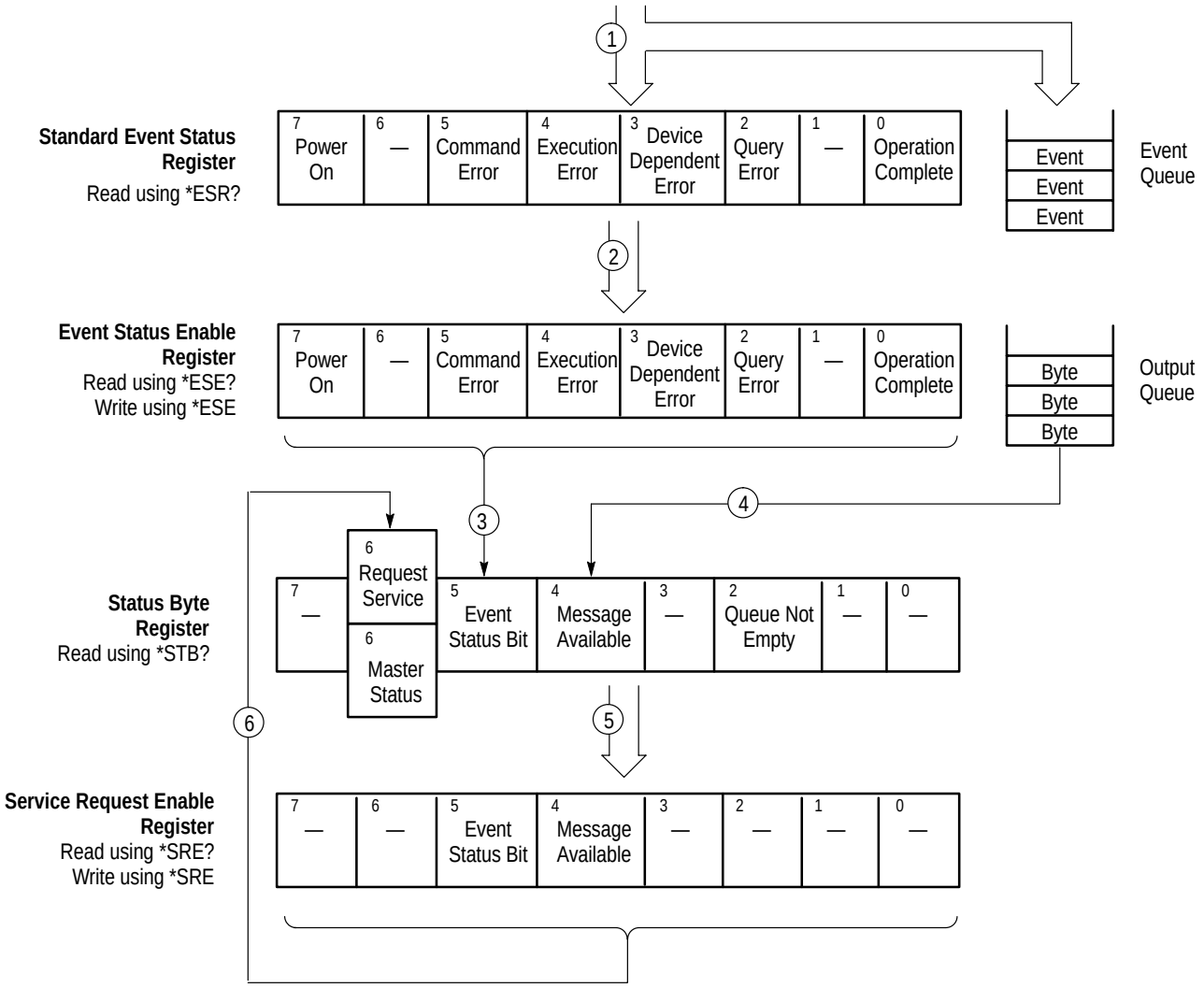


Figure 3–1: Status and Event Reporting Process

When an event occurs the appropriate bit in the Standard Event Status Register is set to one and the event is recorded in the Event Queue (1). If the corresponding bit in the Event Status Enable Register is also enabled (2), then the event status bit in the Status Byte Register is set to one (3).

When output is sent to the Output Queue (for example, a response to a query), the message available bit in the Status Byte Register is set to one (4).

When a bit in the Status Byte Register is set to one and the corresponding bit in the Service Request Enable Register is enabled (5), the master status summary bit in the Status Byte Register is set to one and a service request is generated (6).

Synchronization Methods

Although most GPIB commands are completed almost immediately after being received by the CTS 710, some commands initiate processes requiring additional time. For example, after you send a SENSe:DATA:TELecom:AUTOscan command, you must wait until it has completed execution before you give another command or query.

Sometimes the result of an operation depends on the result of an earlier operation (the first operation must be completed before the next one is initiated). The status and event reporting system of the CTS 710 provides this capability.

Using the *OPC? Query

Use the *OPC? query to synchronize commands. The *OPC? query places a 1 in the Output Queue once an operation is complete. A timeout could occur if you try to read the output queue before there is any data in it.

The same command sequence using the *OPC? query for synchronization looks like this:

```
/* Set up a chained message */
SENSE:DATA:TELECOM:AUTOSCAN;*OPC?
```

Messages

The CTS 710 generates error messages in response to events caused by commands or queries. Each type of event sets a specific bit in the Standard Event Status Register. Thus, each message is associated with a specific Standard Event Status Register bit. In the message tables that follow, the associated Standard Event Status Register bit is specified in the table title. Not shown in the tables are secondary messages giving more detail about the cause of the error or the meaning of the message. These secondary messages are shown for each command and query in *Syntax and Commands*.

Table 3–5 shows the error messages generated by improper command syntax. Check to see that the command is properly formatted and that it follows the rules in *Syntax and Commands*.

Table 3–5: Command Error Messages (Bit 5 in Standard Event Status Register)

Code	Message
100	Command error
101	Invalid character
102	Syntax error
103	Invalid separator
104	Data type error
105	Get not allowed
106	Invalid program data separator
108	Parameter not allowed
109	Missing parameter
110	Command header error
111	Header separator error
112	Mnemonic too long
113	Undefined header
118	Query not allowed
120	Numeric data error
121	Invalid char in number
123	Exponent too large
124	Too many digits
128	Numeric data not allowed
130	Suffix error
131	Invalid suffix
134	Suffix too long

Table 3–5: Command Error Messages (Bit 5 in Standard Event Status Register) (Cont.)

Code	Message
138	Suffix not allowed
140	Character data error
141	Invalid character data
144	Character data too long
148	Character data not allowed
150	String data error
151	Invalid string data
158	String data not allowed
160	Block data error
161	Invalid block data
168	Block data not allowed

Table 3–6 lists the execution error messages that can occur during execution of a command.

Table 3–6: Execution Error Messages (Bit 4 in Standard Event Status Register)

Code	Message
200	Execution error
220	Parameter error
221	Settings conflict
222	Data out of range
223	Too much data
224	Illegal parameter value
230	Data corrupt or stale
240	Hardware error
241	Hardware missing
250	Mass storage error
252	Missing mass storage
252	Missing media
253	Corrupt media
254	Media full
255	Directory full

Table 3–6: Execution Error Messages (Bit 4 in Standard Event Status Register) (Cont.)

Code	Message
256	File name not found
257	File name error
258	Media protected

Table 3–7 lists the device dependent error messages that can occur during CTS 710 operation.

Table 3–7: Device Dependent Error Messages (Bit 3 in Standard Event Status Register)

Code	Message
300	Device specific error
310	System error
361	Autoscan failed

Table 3–8 lists the system events.

Table 3–8: System Events

Code	Message
401	Power on[1]
402	Operation complete[2]

1 Sets bit 7 in the Standard Event Status Register.

2 Sets bit 0 in the Standard Event Status Register.

Table 3–9 lists the execution warnings that can occur during execution of a command.

Table 3–9: Execution Warning Messages (Bit 3 in Standard Event Status Register)

Code	Message
500	Execution warnings

Examples

The following sample program sets up the CTS 710 to perform a one minute bit error rate test. The program is written in Microsoft C and uses a National Instruments GPIB driver. Note that the program verifies communication with the instrument, inserts section code violation errors at a rate of 10^{-5}, loops until the test completes, and prints the BER for the section code violations. Use this program as a basis for programs that perform more advanced tasks. This example program assumes that the instrument is configured to device 4.

```
/*-----------------------------------------------------------------
* Program: BER.C
* Description: This program will set up the instrument to perform
*            a 1 minute BER test. The generator may be looped
*            back to the receiver. Errors will be inserted
*            during the test. The results will be printed at
*            the completion of the test.
* Prerequisites:
*            The instrument must be configured at device 4 in
*            IBCONF.
*            The language is Microsoft C using National
*            Instruments GPIB drivers.
*-----------------------------------------------------------------
*/
#include "stdio.h"
#include "string.h"
#include "decl.h"
main()
{
   char buffer[255];
   int  count = 0;
```

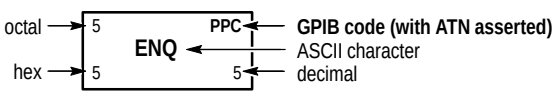
```
int  status = 0;
int device;
/* announce start of program */
printf("CTS 710 BER Program Starting\n");
/* verify instrument is connected */
device = ibfind ("DEV4");
ibclr( device);
buffer[0] = 0;              /* initialize the string */
ibwrt(device,"*IDN?",5);    /* send query to instrument */
ibrd(device, buffer, 255); /* get response from instrument */
buffer[ibcnt-1] = 0;        /* add null to terminate string */
if( ibcnt > 1 )
{
   printf("Instrument at Address 4\n %s\n ", buffer);
}
else
{
   printf("Instrument at Address 4 did not respond");
   return( 1 ); /* error and exit */
}
/* initialize instrument */
ibwrt( device,"SYSTEM:MODE SONET",17);
ibwrt( device "*RST",4);
ibwrt( device "*OPC?",5);
ibrd(device,buffer,255);
/* setup transmitter - B1 errors at 1e-5 rate */
ibwrt( device,"SYSTEM:HEADER 0", 15);
ibwrt( device,"INPUT1:TEL:RATE STS1", 20);
```

```
    ibwrt( device,"OUTPUT1:TEL:RATE STS1", 21);
    ibwrt( device, "SOURCE:DATA:TEL:ERROR:ENABLE ON", 31);
    ibwrt( device, "SOURCE:DATA:TEL:ERROR:TYPE SCV", 30);
    ibwrt( device, "SOURCE:DATA:TEL:ERROR:RATE 1E-5", 31);
    /* setup receiver - test duration of 1 minute */
    ibwrt( device, "SENSE:DATA:TEL:TEST:DURATION 0,0,1,0", 36);
    /* start test */
    ibwrt( device, "SENSE:DATA:TEL:TEST:START", 25);
    /* wait until test is complete */
    do
    {
       ibwrt( device, "SENSE:DATA:TEL:TEST:STATUS?", 27 );
       ibrd( device, buffer, 255 );
       buffer[ibcnt-1] = 0;    /* add null to terminate string */
       printf(">> Elapsed Time =  %s       \r", buffer );
       sscanf( buffer, "%d", &status );
    }while( status == 1 );
    /* query and print results */
    ibwrt( device, "SENSE:DATA:TEL:MEAS:ERROR:ERATIO:SCV?", 37 );
    ibrd( device, buffer, 255 );
    buffer[ibcnt+1] = 0;       /* add null to terminate string */
    printf( "\n>> BER =  %s\n", buffer );
    /* announce end of program */
    printf("End of Test\n");
    /* exit */
    return(0);
}
```

Appendix A: ASCII & GPIB Code Chart

B7 B6 B5 BITS B4 B3 B2 B1	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	CONTROL		NUMBERS SYMBOLS		UPPER CASE		LOWER CASE	
0 0 0 0	0 NUL 0 0	20 DLE 10 16	40 LA0 SP 20 32	60 LA16 0 30 48	100 TA0 @ 40 64	120 TA16 P 50 80	140 SA0 ` 60 96	160 SA16 p 70 112
0 0 0 1	1 GTL SOH 1 1	21 LL0 DC1 11 17	41 LA1 ! 21 33	61 LA17 1 31 49	101 TA1 A 41 65	121 TA17 Q 51 81	141 SA1 a 61 97	161 SA17 q 71 113
0 0 1 0	2 STX 2 2	22 DC2 12 18	42 LA2 " 22 34	62 LA18 2 32 50	102 TA2 B 42 66	122 TA18 R 52 82	142 SA2 b 62 98	162 SA18 r 72 114
0 0 1 1	3 ETX 3 3	23 DC3 13 19	43 LA3 # 23 35	63 LA19 3 33 51	103 TA3 C 43 67	123 TA19 S 53 83	143 SA3 c 63 99	163 SA19 s 73 115
0 1 0 0	4 SDC EOT 4 4	24 DCL DC4 14 20	44 LA4 $ 24 36	64 LA20 4 34 52	104 TA4 D 44 68	124 TA20 T 54 84	144 SA4 d 64 100	164 SA20 t 74 116
0 1 0 1	5 PPC ENQ 5 5	25 PPU NAK 15 21	45 LA5 % 25 37	65 LA21 5 35 53	105 TA5 E 45 69	125 TA21 U 55 85	145 SA5 e 65 101	165 SA21 u 75 117
0 1 1 0	6 ACK 6 6	26 SYN 16 22	46 LA6 & 26 38	66 LA22 6 36 54	106 TA6 F 46 70	126 TA22 V 56 86	146 SA6 f 66 102	166 SA22 v 76 118
0 1 1 1	7 BEL 7 7	27 ETB 17 23	47 LA7 ' 27 39	67 LA23 7 37 55	107 TA7 G 47 71	127 TA23 W 57 87	147 SA7 g 67 103	167 SA23 w 77 119
1 0 0 0	10 GET BS 8 8	30 SPE CAN 18 24	50 LA8 (28 40	70 LA24 8 38 56	110 TA8 H 48 72	130 TA24 X 58 88	150 SA8 h 68 104	170 SA24 x 78 120
1 0 0 1	11 TCT HT 9 9	31 SPD EM 19 25	51 LA9) 29 41	71 LA25 9 39 57	111 TA9 I 49 73	131 TA25 Y 59 89	151 SA9 i 69 105	171 SA25 y 79 121
1 0 1 0	12 LF A 10	32 SUB 1A 26	52 LA10 * 2A 42	72 LA26 : 3A 58	112 TA10 J 4A 74	132 TA26 Z 5A 90	152 SA10 j 6A 106	172 SA26 z 7A 122
1 0 1 1	13 VT B 11	33 ESC 1B 27	53 LA11 + 2B 43	73 LA27 ; 3B 59	113 TA11 K 4B 75	133 TA27 [5B 91	153 SA11 k 6B 107	173 SA27 { 7B 123
1 1 0 0	14 FF C 12	34 FS 1C 28	54 LA12 , 2C 44	74 LA28 < 3C 60	114 TA12 L 4C 76	134 TA28 \ 5C 92	154 SA12 l 6C 108	174 SA28 ¦ 7C 124
1 1 0 1	15 CR D 13	35 GS 1D 29	55 LA13 – 2D 45	75 LA29 = 3D 61	115 TA13 M 4D 77	135 TA29] 5D 93	155 SA13 m 6D 109	175 SA29 } 7D 125
1 1 1 0	16 SO E 14	36 RS 1E 30	56 LA14 . 2E 46	76 LA30 > 3E 62	116 TA14 N 4E 78	136 TA30 ^ 5E 94	156 SA14 n 6E 110	176 SA30 ~ 7E 126
1 1 1 1	17 SI F 15	37 US 1F 31	57 LA15 / 2F 47	77 UNL ? 3F 63	117 TA15 O 4F 79	137 UNT – 5F 95	157 SA15 o 6F 111	177 RUBOUT (DEL) 7F 127
	ADDRESSED COMMANDS	UNIVERSAL COMMANDS	LISTEN ADDRESSES		TALK ADDRESSES		SECONDARY ADDRESSES OR COMMANDS	

KEY



Tektronix
REF: ANSI STD X3.4-1977
IEEE STD 488.1-1987
ISO STD 646-2973

Appendix B: Commands and Queries Posting OPC

The commands and queries listed in Table B–1 set the OPC bit after execution. Some of these commands and queries may require more than 200 ms to complete execution.

Table B–1: Commands and Queries that Post OPC

Command or Query
*RST
*RCL
*SAV
*TST?
HCOPy:IMMediate
DIAGnostic:EXECute
all MMEMory: commands
INPUT1:TELecom:TYPE
SOURce:CLOCk:SOURce
SENSe:DATA:TELecom:AUTOscan

Appendix C: *LRN? Response

Table C–1 lists the commands that are returned by the *LRN? and SYSTem: SET? queries. The response consists of these commands and their parameter values separated by semicolons (;).

Table C–1: Commands Returned by *LRN?

Command
SYSTem:MODE
SYSTem:SDEScription
SYSTem:BEEPer:CONTrol
SYSTem:HEADers
SYSTem:VERBose
OUTPUT1:TELecom:RATE
OUTPUT1:TELecom:TYPE
OUTPUT1:TELecom:LEVel
OUTPUT2:TELecom:TERMinator (Add/Drop/Test Option Only)
OUTPUT2:TELecom:CODE (Add/Drop/Test Option Only)
OUTPUT3:TELecom:RATE (Add/Drop/Test Option Only)
SOURce:CLOCk:SOURce
SOURce:CLOCk:OFFSet:MODE
SOURce:CLOCk:OFFSet:LVALue
SOURce:CLOCk:OFFSet:PVALue (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:SOURce
SOURce:DATA:TELecom:STRUcture
SOURce:DATA:TELecom:CHANnel
SOURce:DATA:TELecom:PAYLoad:MAPPing
SOURce:DATA:TELecom:PAYLoad:PATTern
SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe
SOURce:DATA:TELecom:OVERhead:APS
SOURce:DATA:TELecom:OVERhead:INSert
SOURce:DATA:TELecom:POVerhead:INSert
SOURce:DATA:TELecom:POVerhead:TRAce
SOURce:DATA:TELecom:ERRor:ENABle

Table C–1: Commands Returned by *LRN? (Cont.)

Command
SOURce:DATA:TELecom:ERRor:TYPE
SOURce:DATA:TELecom:ERRor:RATE
SOURce:DATA:TELecom:ALARm
SOURce:DATA:TELecom:FAILure:TYPE
SOURce:DATA:TELecom:POINter:MODE
SOURce:DATA:TELecom:POINter:NDFlag
SOURce:DATA:TELecom:POINter:DIRection
SOURce:DATA:TELecom:POINter:RATE
SOURce:DATA:TELecom:POINter:NBURst
SOURce:DATA:TELecom:POINter:SBITs
SOURce:DATA:TELecom:POINter:SEQuence:TYPE
SOURce:DATA:TELecom:POINter:SEQuence:DIRection
SOURce:DATA:TELecom:POINter:SEQuence:IPERiod
SOURce:DATA:TELecom:POINter:SEQuence:CPERiod
SOURce:DATA:TELecom:POINter:SEQuence:RATE
SOURce:DATA:TELecom:TRIButary:CHANnel (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:MAPPing (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:FRAMing (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:PATTern (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:PATTern:UWORD (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:BACKground:PATTern (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:ADD (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POVerhead:TRACe (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:ERRor (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:ALARm (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:FAILure (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:MODE (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:NDFLag (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:DIRection (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:RATE (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:NBURst (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE (Add/Drop/Test Option Only)

Table C–1: Commands Returned by *LRN? (Cont.)

Command
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:DIRection (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE (Add/Drop/Test Option Only)
INPUT1:TELecom:RATE
INPUT1:TELecom:TYPE
INPUT1:TELecom:LEVel
INPUT2:TELecom:TERMinator (Add/Drop/Test Option Only)
INPUT2:TELecom:LEVel (Add/Drop/Test Option Only)
INPUT3:TELecom:RATE (Add/Drop/Test Option Only)
INPUT3:TELecom:LEVel (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:SOURce
SENSe:DATA:TELecom:STRUcture
SENSe:DATA:TELecom:CHANnel
SENSe:DATA:TELecom:PAYLoad:MAPPing
SENSe:DATA:TELecom:PAYLoad:PATTern
SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe
SENSe:DATA:TELecom:TRIButary:CHANnel (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:MAPPing (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:FRAMing (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:CHANnel (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:DEMULtiplex:FRAMing (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:PATTern (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:DROP (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TEST:DURation
SENSe:DATA:TELecom:TEST:HISTory:RESolution
SENSe:DATA:TELecom:OVERhead:DROP
SENSe:DATA:TELecom:POVerhead:DROP
INSTrument:COUPling
HCOPY:ITEM

Table C–1: Commands Returned by *LRN? (Cont.)

Command
HCOPy:TITle
HCOPy:DEVice:LANGuage

Appendix D: Default Parameter Values After *RST

Table D–1 lists the default parameter values after the *RST command is sent.

Table D–1: Default Parameter Values After *RST

Command	Default Parameter Values
OUTPUT1:TELecom:RATE	STS1
OUTPUT1:TELecom:TYPE	ELECtrical
OUTPUT1:TELecom:LEVel	XCONnect
OUTPUT2:TELecom:TERMinator	BALanced (Add/Drop/Test Option Only)
OUTPUT2:TELecom:CODE	AMI (Add/Drop/Test Option Only)
OUTPUT3:TELecom:RATE	DS3 (Add/Drop/Test Option Only)
SOURce:CLOCk:SOURce	INTernal
SOURce:CLOCk:OFFSet:MODE	LOFFset
SOURce:CLOCk:OFFSet:LVALue	0.00
SOURce:CLOCk:OFFSet:PVALue	0.00
SOURce:DATA:TELecom:SOURce	OUTPUT1
SOURce:DATA:TELecom:STRUcture	STS1
SOURce:DATA:TELecom:CHANnel	1
SOURce:DATA:TELecom:PAYLoad:MAPPing	EQUIPPED
SOURce:DATA:TELecom:PAYLoad:PATTern	PRBS23
SOURce:DATA:TELecom:PAYLoad:PATTern:UBYTe	0
SOURce:DATA:TELecom:OVERhead:PRESet	(sets overhead to values listed in Figure 2–14 in *Syntax and Commands* on page 2–38)
SOURce:DATA:TELecom:OVERhead:APS	0
SOURce:DATA:TELecom:OVERhead:INSert	NONE
SOURce:DATA:TELecom:POVerhead:INSert	NONE
SOURce:DATA:TELecom:POVerhead:TRAce	“ ” (64 null characters)
SOURce:DATA:TELecom:ERRor:ENABle	0
SOURce:DATA:TELecom:ERRor:TYPE	SCV
SOURce:DATA:TELecom:ERRor:RATE	1E–10
SOURce:DATA:TELecom:ALARm	NONE
SOURce:DATA:TELecom:FAILure:TYPE	NONE

Table D–1: Default Parameter Values After *RST (Cont.)

Command	Default Parameter Values
SOURce:DATA:TELecom:POINter:MODE	SINGle
SOURce:DATA:TELecom:POINter:VALue	522
SOURce:DATA:TELecom:POINter:NDFlag	1
SOURce:DATA:TELecom:POINter:DIRection	ALTernate
SOURce:DATA:TELecom:POINter:RATE	100
SOURce:DATA:TELecom:POINter:NBURst	2
SOURce:DATA:TELecom:POINter:SBITs	0
SOURce:DATA:TELecom:POINter:Sequence:TYPE	SINGle
SOURce:DATA:TELecom:POINter:Sequence:DIRection	UP
SOURce:DATA:TELecom:POINter:Sequence:IPERiod	1
SOURce:DATA:TELecom:POINter:Sequence:CPERiod	1
SOURce:DATA:TELecom:POINter:Sequence:RATE	30000
SOURce:DATA:TELecom:TRIButary:CHANnel	1 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:MAPPing	VTASYNC (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:FRAMing	UNFRamed (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:PATTern	PRBS23 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:PATTern:UWORd	0 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:PATTern: UWORd:LENgth	1 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:BACKground: PATTern	QRSS (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:ADD	0 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POVerhead:TRACe	TEK CTS750 (Performs no operation
SOURce:DATA:TELecom:TRIButary:ERRor	NONE (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:ALARM	NONE (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:FAILure	NONE (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:MODE	SINGle (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:NDFLag	1 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:DIRection	ALTernate (Add/Drop/Test Option Only)

Table D–1: Default Parameter Values After *RST (Cont.)

Command	Default Parameter Values
SOURce:DATA:TELecom:TRIButary:POINter:RATE	100 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:Nburst	2 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:TYPE	SINGle (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:DIRection	UP (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:IPERiod	1 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:CPERiod	1 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:POINter:SEQuence:RATE	30000 (Add/Drop/Test Option Only)
SOURce:DATA:TELecom:TRIButary:MODE	MANual (Add/Drop/Test Option Only)
INPUT1:TELecom:RATE	STS1
INPUT1:TELecom:TYPE	ELECtrical
INPUT1:TELecom:LEVel	XCONnect
INPUT2:TELecom:TERMinator	BALanced (Add/Drop/Test Option Only)
INPUT2:TELecom:LEVel	NORMal (Add/Drop/Test Option Only)
INPUT3:TELecom:RATE	DS3 (Add/Drop/Test Option Only)
INPUT3:TELecom:LEVel	NORMal (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:SOURce	INPUT1
SENSe:DATA:TELecom:STRUcture	STS1
SENSe:DATA:TELecom:CHANnel	1
SENSe:DATA:TELecom:PAYLoad:MAPPing	EQUIPPed
SENSe:DATA:TELecom:PAYLoad:PATTern	PRBS23
SENSe:DATA:TELecom:PAYLoad:PATTern:UBYTe	0
SENSe:DATA:TELecom:TRIButary:CHANnel	1 (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:MAPPing	VTASYNC (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:FRAMing	UNFRamed (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:Demultiplex:CHANnel	1 (Add/Drop/Test Option Only)

Table D–1: Default Parameter Values After *RST (Cont.)

Command	Default Parameter Values
SENSe:DATA:TELecom:TRIButary:Demultiplex:FRAMing	UNFramed (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:PATTern	PRBS23 (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd	0 (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:PATTern:UWORd:LENgth	1 (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TRIButary:DROP	0 (Add/Drop/Test Option Only)
SENSe:DATA:TELecom:TEST:DURation	0,0,0,0
SENSe:DATA:TELecom:TEST:HISTory:RESolution	MIN1
SENSe:DATA:TELecom:OVERhead:DROP	NONE
SENSe:DATA:TELecom:POVerhead:DROP	NONE
SENSe:DATA:TELecom:MEASure:STESts:	(all four commands set to NONE)
INSTrument:COUPling	NONE
INITiate	(executed)
SYSTem:BEEPer:CONTrol	0
SYSTem:HEADers	1 (ON)
SYSTem:MODE	SONET
SYSTem:SDEScription	” ”
SYSTem:VERBose	1 (ON)
DISPlay:BRIGhtness	1.00
HCOPy:DEVice:LANGuage	TEK
HCOPy:ITEM	SUMMary
HCOPy:TITle	0

Glossary

AIS
An acronym for Alarm Indication Signal. An AIS is used to alert downstream equipment that an upstream failure has been detected.

APS
An acronym for Automatic Protection Switching.

ASCII
An acronym for American Standard Code for Information Interchange.

BER
An acronym for Bit Error Ratio (or Rate). The principal measure of quality of a digital transmission system. BER is defined as:

$$BER = \frac{Number\ of\ Errors}{Total\ Number\ of\ Bits}$$

BER is usually expressed as a negative exponent. For example, a BER of 10^{-7} means that 1 bit out of 10^{7} bits is in error.

BIP
An acronym for Bit Interleaved Parity. A method used to monitor errors in the transmitted signal.

Bit Error
An incorrect bit. Also known as a coding violation.

BITS
An acronym for Building Integrated Timing Supply.

Chained Commands and Queries
Commands and queries grouped together into a single message to be sent to the instrument. Each command and query in the chained message must be separated by a semicolon (;).

COFA
An acronym for Change of Frame Alignment.

Command
Specifies an action for the instrument to perform.

Common Commands
See IEEE 488.2 Common Commands.

CV
An acronym for Coding Violation.

dB
The symbol for decibels.

dBm
The symbol for power level in decibels relative to 1 mW.

DM
An acronym for Degraded Minute. A minute with a BER greater than 10^{-6}.

DSn
An acronym for Digital Signal-n (DS1, DS2, DS3, and DS4). DS1 is the basic multiplex rate in North America; additional rates are DS2, DS3, and DS4. The following table lists the DSn rates and their multiple of DS1:

Digital Signal Transmission Rates

Level	Rate	Multiple of DS1
DS1	1.544 Mb/s	1
DS1c	3.152 Mb/s	2
DS2	6.312 Mb/s	4
DS3	44.736 Mb/s	24

EFS
An acronym for Error Free Seconds.

Equipped
An interpretation of mapping; the C2 byte in the path overhead is set to one.

Errored Seconds – Type A
The number of seconds that contain only one CV and no defects.

Errored Seconds – Type B
The number of seconds that contain more than one and less than N CVs and no defects. The range for N depends on the transmit rate, but corresponds to a BER of about 10^{-6}.

ES
An acronym for Errored Second. A second with at least one error.

Event Status Enable Register
The register that controls which events are summarized in the event status bit (bit 5) of the Status Byte Register.

FEBE
An acronym for Far End Block Error. An indication returned to the transmitting LTE that an errored block has been detected at the receiving LTE.

FERF
An acronym for Far End Receive Failure. A FERF indicates to the transmitting LTE that the receiving LTE has detected an incoming line failure or is receiving a Line AIS.

Functional Block
A part of the instrument that performs a specific function. Examples are the Transmitter and Receiver.

IEEE 488.21 Common Commands
The set of commands and queries defined by the ANSI/IEEE Standard 488.2.

ITU
An acronym for the International Telecommunication Union.

Line
The portion of a transmission line between two multiplexers.

Line Alarm Indication Signal (AIS)
A Line AIS is generated by Section Terminating Equipment upon Loss of Signal or Loss of Frame.

Line Coding Violation (CV)
The sum of the BIP errors detected at the Line layer. Line CVs are collected using the BIP codes in the B2 bytes of the Line Overhead.

Line Errored Second (ES)
A second during which at least one Line CV occurred, or a second during which the line was in the Line AIS state.

Line Far End Receive Failure (FERF)
An indication returned to a transmitting LTE from the receiving LTE that a Line AIS or incoming line failure has been detected.

Line Overhead (LOH)
Controls the payload information using the section layer and provides alarm indications, error monitoring, and message signalling between two LTEs.

Line Severely Errored Second (SES)
A second with N or more Line CVs, or a second during which the line was in the Line AIS state. The value of N varies with the transmit rate, but corresponds to a 2×10^{-7} BER.

LOF
An acronym for Loss of Frame.

LOP
An acronym for Loss of Pointer.

LOS
An acronym for Loss of Signal.

LTE
An acronym for Line Terminating Equipment.

Mapping
The process of placing a tributary signal into a SONET SPE.

Mb/s
Megabits per second.

NE
An acronym for Network Element.

Normal Mode
A mode of operation in which the Transmitter and the Receiver operate independently.

OC
An acronym for Optical Carrier.

OOF
An acronym for Out of Frame.

OPC Bit (Operation Complete Bit)
A bit in the Standard Event Status Register that is set when certain commands and queries complete execution.

Optical Carrier Level N (OC-N)
An optical version of an STS-N signal.

Output Queue
The queue that stores query responses from the instrument.

Pass/Fail Test
A test measurement that is compared to predefined thresholds to determine a pass or fail condition of the test.

Path
The portion of a transmission network between two terminal multiplexers.

Path Overhead (POH)
A set of bytes allocated within the information payload to carry status and maintenance information between two network elements.

Path Trace
A 64-character sequence used to identify the communication path. The path trace is carried in the J1 byte over 64 repeating frames.

POH
An acronym for Path Overhead.

PTE
An acronym for Path Terminating Equipment.

Query
A message sent to the instrument that returns information about the state of the instrument.

Receiver
The functional block of the instrument that measures an incoming signal.

Rx
An abbreviation for Receive.

SCPI
An acronym for Standard Commands for Programmable Instruments. A standard that provides guidelines for remote programming of instruments.

Section
The portion of a transmission line between a Network Element (NE) and a Line Terminating Equipment (LTE) or two LTEs.

Section Coding Violation (CV)
A BIP error that is detected at the Section layer. CVs for the Section layer are collected using the BIP-8 in the B1 byte located in the Section overhead of STS-1 number 1.

Section Errored Second (ES)
A second during which at least one Section CV or OOF/COFA event occurred, or a second during which the NE was (at any point during the second) in the LOS state.

Section Overhead (SOH)
A set of bytes allocated within each frame to carry framing and error monitoring information between an NE and LTE or between two LTEs. Part of the transport overhead.

Service Request Enable Register
The register that controls which bits in the Status Byte Register generate a service request

SES
An acronym for Severely Errored Seconds.

Severely Errored Seconds (SES)
A second with more than N CVs. N varies with the transmit rate but corresponds to a BER of $2x10^{-7}$.

SOH
An acronym for Section Overhead.

SONET
An acronym for Synchronous Optical NETwork.

SPE
An acronym for Synchronous Payload Envelope.

Standard Event Status Register
The register that records many types of events that occur in the instrument such as execution error and operation complete.

Status Byte Register
The register that summarizes information from other registers in the Status and Event Reporting System.

STE
An acronym for Section Terminating Equipment.

STS
An acronym for Synchronous Transport Signal.

STS-N
An acronym for Synchronous Transport Signal level-N (STS-1, -3, -9, -12, -18, -24, -36, -48). The different STS-N rates (and their optical equivalents) for the SONET Signal Hierarchy are listed in the following table:

SONET Signal Hierarchy

Electrical Signal	Optical Signal	Data Rate (Mb/s)
STS-1	OC-1	51.84
STS-3	OC-3	155.52
STS-9	OC-9	466.56
STS-12	OC-12	622.08
STS-18	OC-18	933.12
STS-24	OC-24	1244.16
STS-36	OC-36	1866.24
STS-48	OC-48	2488.32

Subsystem Hierarchy Tree
A graphical representation of a subsystem of commands and queries.

System Error and Event Queue
The queue that stores error and event messages.

TE
An acronym for Terminal Equipment.

Through Mode
The ability to retransmit the incoming signal and manipulate its contents.

TOH
An acronym for Transport Overhead.

Transmitter
The functional block of the instrument that generates a signal.

Transport Overhead (TOH)
A set of bytes allocated within each frame to carry operation, administration, and maintenance information from one end of the system to the other.

Tributary
The lower rate signal that is input to a multiplexer for combination (multiplexing) with other low rate signals to form a higher rate signal.

Tx
An abbreviation for Transmit.

Unequipped
An interpretation of mapping; the C2 byte in the path overhead is set to zero.

VT
An acronym for Virtual Tributary. A structure (not a signal) designed for transport and switching of sub-STS payloads. The sizes of VT currently in use are VT1.5, VT2, VT3, and VT6.

Yellow Signal
A code sent upstream to indicate that a failure condition has been declared downstream.

Index

L

M

O

P

Q

R

S

T

W