Programmer Manual

p

ST2400A
2.4 Gb/s SDH/SONET Test Set

071-0243-03

This document supports firmware version 5.0 and above.

Tektronix, Inc., P.O. Box 1000, Wilsonville, OR 97070-1000

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 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 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 user 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.

How to Reach Customer Service

If you have any questions regarding the operation, maintenance, repair, or application of your Tektronix equipment, contact your local sales and service office. For a complete list of the Worldwide Sales and Service Offices contact (800) 426-2200.

Tektronix provides high quality Technical Support on applications, operation, measurement specifications, hardware, and software by expert application engineers. For Applications Support, call the Customer Support Center listed below.

Mailing Address	Tektronix, Inc. Measurement Business Division P.O. Box 500 Beaverton, Oregon 97077-0001 USA	Attn. Customer Service
Customer and Sales Support Center	**800-TEK-WIDE** **or** **800-835-9433 Ext 2400**	Hours are 6:00 AM to 5:00 PM, Pacific Time. After hours Voice Mail is available.
Direct	**503-627-2400**	
Fax	**503-627-5695**	
E-Mail	tm_app_supp@tek.com	
Web Site	http://www.tek.com	

U.S.A., Africa, Asia, Australia, Central & South America, Japan
Tektronix, Inc.
P.O. Box 500
Beaverton, Oregon 97077-0001
For additional literature, or the address and phone number of the Tektronix Sales Office or Representative nearest you,
contact: (800) 426-2200

Belgium: Brusells
Phone: 32(2) 725 96 10
Fax: 32(2) 725 99 53

Canada: Calgary
Phone: (403) 274-2691
Fax: (403) 274-3483

Denmark: Copenhagen
Phone: 45 44 850 700
Fax: 45 44 850 701

Eastern Europe, Middle East and Austria
Tektronix Ges.m.b.H
Triester Strasse 14
A-2351 Wiener Neudorf, Austria
Phone: ++43 (2236) 8092-0
Fax: ++43 (2236) 8092-200

Finland: Helsinki
Phone: 358 4783 400
Fax: 358 47834 200

France and North Africa
Tektronix S.A.
ZAC Courtaboeuf, 4 Av du Canada, B.P.13
91941 Les Ulis Cedex, France
Phone: 33(1) 69 86 81 81
Fax: 33(1) 69 07 09 37

Germany: Cologne
Phone: 49(221) 94770
Fax: 49(221) 9477 200

Italy: Milan
Phone: 39(2) 25 0861
Fax: 39(2) 25 086 400

Japan: Tokyo
Phone: 81(3) 3448-3111
Fax: 81(3) 3444-3661

The Netherlands: Hoofddorp
Phone: 31(23) 069 5555
Fax: 31(23) 569 5500

Norway: Oslo
Phone: 47(22) 07 0700
Fax: 47(22) 07 0707

Spain: Madrid
Phone: 34(1) 372 6000
Fax: 34(1) 372 6049

Sweden: Stockholm
Phone: 46(8) 629 6500
Fax: 46(8) 629 6540

Switzerland: Zug
Phone: 41(42) 219192
Fax: 41(42) 217784

U.K.: Marlow

Tektronix sales and service
offices around the world:
Algeria,
Argentina, Australia,
Bahrain, Bangladesh,
Belgium,
Bolivia,
Brazil,
Bulgaria,
Canada,
Chile,
People’s Republic
of China, Columbia,
Costa Rica, Cypress,
Czechoslovakia, Denmark,
Ecuador,
Egypt,
Finland,
France,
Germany,
Greece,
Hong Kong, Iceland,
India,
Indonesia,
Ireland,
Israel,
Italy,
Ivory Coast,
Japan,
Jordan,
Korea,
Kuwait,
Lebanon, Malaysia,
Mexico,
The Netherlands, New
Zealand, Nigeria,
Norway,
Oman,
Pakistan,
Panama,
Peru,
Phillippines, Poland,
Portugal,
Saudi Arabia, South Africa,
Singapore,
Spain,
Sri Lanka,
Sweden, Switzerland, Taiwan,
Thailand,
Tunisia,
Turkey,
United Arab Emirates,
United Kingdom,
Uruguay,
Venezuela,
Zimbabwe

Phone: 44 1628 403300
Fax: 44 1628 403301

Table Of Contents

EXTERNAL CONTROLLERS & PRINTERS

REMOTE COMMANDS

APPENDICES

General Safety Summary

Review the following safety precautions to avoid injury and prevent damage to this product or any equipment connected to it. To avoid potential hazards, use this product only as specified.

To avoid potential hazards, use this product only as specified.

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 protections if there is a possibility of exposure to high-intensity rays.

Product Damage Precautions

⚠ CAUTION!

Signal levels greater than –10dBm (negative 10dBm) may damage the Optical Input devices. Always pad the input level to less than –10dBm.

Always use 15 dB of attenuation when connecting the ST2400A Transmitter output to its receiver input. Failure to do so will damage the optical detector.

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:

Icon	Label	Meaning
⚠	**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 produce 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 American power network. All other power cords supplied are approved for the country of use.

Preface

The ST2400A can be controlled remotely through the use of Remote Commands. This manual describes how to use these commands to access information generated by or stored in the instrument.

How This Manual is Organized

This manual is divided into three sections: *External Controllers and Printers; Remote Commands; and Appendices*.

- *External Controllers and Printers* explains how to connect external devices to the RS-232C, GPIB, and parallel ports. This section also includes port setup procedures.
- *Remote Commands* explains the general syntax of the ST2400A remote command language and defines all commands.
- *Appendices* presents Glossary and Reference material for the SDH and the SONET telecommunications technologies.

Conventions

This manual uses the following conventions:

- The names of front-panel controls and menus appear in all upper case letter, for example, TRANSMIT and HELP.
- Names appear in the same case in this manual as they appear on the display of the ST2400A , for example, Test Duration and USER DEFINED.
- Within a procedure, a specific button to be pressed or a parameter to be selected appears in boldface print.

Related Manuals

The following documents are available for the ST2400A SDH/SONET Test Set:

- *ST2400A SDH User Manual* (Tektronix part number 071-0241-00) is the primary source of information on how the ST2400A SDH Test Set functions.
- *ST2400A SONET User Manual* (Tektronix part number 071-0242-00) is the primary source of information on how the ST2400A SONET Test Set functions.
- *ST2400A SONET/SDH Test Set Service Manual* (Tektronix part number 071-0244-00) provides information on maintaining and servicing your instrument to the module level.

External Controllers & Printers

External Controllers and Printers

This chapter explains how to connect external devices to the RS-232C, GPIB, and parallel ports. This chapter also includes port setup procedures.

Remote Control Applications

The **ST2400A 2.4 Gb/s SDH/SONET Test Set** can be remotely operated from its RS-232C or GPIB ports in one of the following ways:

- Use the RS-232C Port manually from a VT-100 terminal or VT-100 emulation program, entering commands individually. Recommended for learning commands and syntax.
- Use the GPIB or RS-232C Port Automatically from a PC or engineering workstation using a control program. Recommended for factory automation applications.

Printer Applications

The **ST2400A 2.4 Gb/s SDH/SONET Test Set** can print reports to its:

- Parallel port using a Centronics printer interface
- Serial port using an RS-232C serial printer interface

Using the RS-232C Port

The following section describes the RS-232C serial port.

Serial Interface Parameters

ST2400A serial port parameters are listed below along with their available values and default value. In addition, ECHO and XON/XOFF can be changed remotely via the GPIB or RS-232 port.

RS-232C (Serial Port) Parameters

Parameter MENU and Description	Possible Values	Default Value
BAUD: Baud rate in bits per second.	300, 1200, 2400, or 9600	9600
PARITY: Error detection.	Even, Odd, or None	None
SIZE: Data bits per character.	7 or 8	8
EOL: End-of-Line terminator.	CR (ASCII decimal 13), LF (ASCII decimal 10), CR/LF, or LF/CR.	CR/LF
XON/XOFF: Flow control.	ON (enabled) or OFF (disabled)	ON
ECHO: Echo back to the controller each character received by the ST2400A).	ON (enabled) or OFF(disabled)	ON

RS-232C Connector Pinout

The RS-232C interface is a 9-pin D-type socket connector located on the rear-panel. The pin-out of this connector is shown below. The ST2400A RS-232C port is wired as a DCE (data communications equipment) device—it receives on pin 2 and transmits on pin 3.

RS-232C Connector Pin-out

PIN	NAME	FUNCTION
1.	GND	Protective Ground
2.	RxD	Received Data Input: Data is received by the ST2400A on this pin.
3.	TxD	Transmit Data Output: Data is transmitted by ST2400A on this pin.
4	CTS	Clear To Send. A high level or open applied to this pin indicates to the ST2400A that the controller or terminal is ready to receive data. A low level indicates that the controller or terminal is not ready. The ST2400A pulls this pin high internally with 27 kohms to +12V.
5.	RTS	Request to Send. The ST2400A always sets this pin high.
6.	DTR	Data Terminal Ready. The ST2400A always sets this pin high.
7.	GND	Signal Ground.
?	DSR	Data Set Ready. This input is ignored.

all other pins	not used.

Cables

Use a straight-through (modem) cable when connecting the ST2400A to a DTE (data terminal equipment) device, or a null-modem cable when connecting to another DCE device.

Quick Test, RS-232C Port

You can perform a quick test of the ST2400A RS-232C port using a terminal (or a PC running terminal emulation software) as follows:

1. Verify that the ST2400A echo mode is on.

Power off the ST2400A .

1. Attach the appropriate 9-pin cable from the RS-232C port to the terminal. Terminals and controller should be configured as a DTE and therefore use straight-through (modem) cable.
2. Power on the ST2400A and verify that the following prompt appears on the terminal's display:

   ST2400A >

If the above prompt does not appear, then one or more of the following problems may exist:

1. The cable may be defective.
2. The cable could be the wrong cable type. Use a straight-through (modem) cable if the terminal is configured as a DTE, or a null-modem cable if the terminal is DCE configured.
3. The RS-232C port setup (baud rate, data bits, parity, etc.) may not be compatible with that of the terminal; both setups should be the same.

Protocol

The ST2400A RS-232C port operates in a terminal mode. Received characters are stored in the receive buffer until a line terminator (CR, LF, LF/CR, or CR/LF) is received, at which point the command is executed. Before a line terminator is received, the backspace key is used to erase the last (right-most) character in the receive buffer. You may use EVEN or ODD parity to perform byte-by-byte error detection.

Command Format

Except for terminator characters, the same commands are used on the GPIB port.

Error Messages

The ST2400A verifies all commands received on the RS-232C port to make sure that they contain the appropriate mnemonics and parameters. All valid commands will be executed. The ST2400A will issue an error message and discards the command when the mnemonic is invalid or an associated parameter is out of range. Error messages are listed below.

RS-232C Port Error Messages

ERROR MESSAGE	MEANING
*** Input Lost	The RS-232C interface hardware has detected one or more lost or corrupted characters.
*** Input Buffer Overflow	More than 80 characters are received without a line terminator.
*** Command Mnemonic Not Found	Unknown command mnemonic received.
*** Invalid Command For Interface	Command mnemonic found, but it is not valid for the RS-232C interface.
*** Invalid Command Type	Command mnemonic found, but it has a missing or added "?" at the end.
*** Too Few Parameters	One or more expected parameters are missing.
*** Too Many Parameters	Too may parameters were received, or unknown characters were found trailing the command.
*** Invalid Parameter	Parameter is not valid for the command.
*** Parameter Out of Range	Parameter value is not in the range specified for the command.
*** Parameter Not In Set	Parameter value is not in the set of allowed values for the command.
*** Invalid String Length	Parameter character string is too long for the command.
*** Parameter Separator	An expected semicolon (;) between two parameters is missing or a command line is terminated immediately after a semicolon.
*** Invalid Non-decimal Parameter	Parameter is not in a valid non-decimal format, or the value of the parameter is out of the range 0 to 255 decimal.
*** Command Execution Error	The instrument did not execute the command properly.
*** Out of Memory	Processor is out of memory.

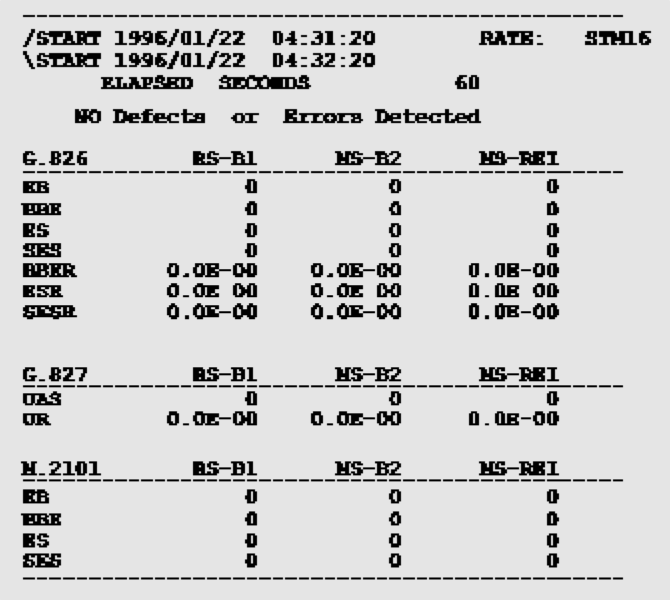
Storing and Printing Data

Most Terminal Emulation packages have the ability to capture data to a file or direct it to a printer. These functions are usually listed as *File Capture* and *Print Capture*. The file capture function prompts for a path and file name before saving the file to disk. saved data files can be reviewed on screen or printed on paper at any time when the data is needed.

Using the RS-232C for PC-based Printing

Enter CTRL + Q to quit VFP mode if it is active. Sequentially push the SETUP key on the ST2400A until the REPORT/TEST TIME menu is displayed. Enter the menu and set up the report as if it was going to be printed on a dedicated printer. Rather than select the default parallel printer port, however, select the RS-232C port instead.

With the RS-232 port and your emulator set up as described earlier, all reports that normally would go to the printer will be displayed on the screen. The scrolling buffer and data capture capabilities provided by your communications software are available for viewing and reviewing the data. An example of the on-screen printer display is shown in the figure below.



```
----------------------------------------------
/START 1996/01/22  04:31:20        RATE:   STM16
\START 1996/01/22  04:32:20
      ELAPSED  SECONDS          60

    NO Defects  or  Errors Detected

G.826         RS-B1      MS-B2      MS-REI
----------------------------------------------
EB                0          0           0
BBE               0          0           0
ES                0          0           0
SES               0          0           0
BBER        0.0E-00    0.0E-00     0.0E-00
ESR         0.0E 00    0.0E 00     0.0E 00
SESR        0.0E-00    0.0E-00     0.0E-00


G.827_________RS-B1______MS-B2______MS-REI_____
UAS               0          0           0
UR          0.0E-00    0.0E-00     0.0E-00

M.2101________RS-B1______MS-B2______MS-REI_____
EB                0          0           0
BBE               0          0           0
ES                0          0           0
SES               0          0           0
----------------------------------------------
```

Figure 1-1. Error Reports Available (Example Screen)

Be sure that the ST2400A printer function is turned off before activating the VT100 emulator. The printer and VT100 cannot work simultaneously through the single ST2400A RS-232 port. If a parallel printer is connected to the ST2400A parallel printer port, then the printer and VT100 remote screen can function simultaneously.

Using the GPIB Port

The following section describes the GPIB port.

GPIB Address and Terminator

Use the Auxiliary Setup menu (shown in *Chapter 3- Reference* of the User Manual) to set the GPIB address and message terminator. They can be set as follows:

- Address from 0 to 31 (Address 31 is off the bus)
- Message terminator is set to EOI or EOI/LF

GPIB Remote and Local Modes

The GPIB controllers can be used to put the ST2400A into a remote mode, disabling all front panel setup controls. The **REM** LED (left side corner of front panel) will indicate whether the ST2400A is in the remote mode (LED on) or the normal local mode (LED off). Press the SETUP key to return the instrument to the local mode.

Local Lock Out

GPIB controllers can also put the ST2400A into the local lock out mode. The **LLO** LED (left side corner of front panel) indicates whether the ST2400A is in the local lock out mode (LED on) or not (LED off). In the LLO mode, like the REMOTE mode, the front panel is disabled. However unlike the REMOTE mode, there is no key that can disable the LLO mode. Instead, you must issue the appropriate GPIB command, or power off and then on the ST2400A .

GPIB Interface Functions

The ST2400A is configured as a GPIB talker/listener. It does not provide any controller functions. The following table lists the GPIB functions described in the IEEE Std. 488.2-1987 that are supported by the ST2400A.

GPIB Interface Functions

Subset	Implementation
SH1	Complete source handshake.
AH1	Complete acceptor handshake.
T6	Basic talker, serial poll, no talk-only, unaddressed if addressed to listen, no extended talker.
L4	Basic listener, no listen-only, unaddressed if addressed to talk, no extended listener.
SR1	Complete service request.
RL1	Remote/local capability including local lockout.
PP0	No parallel poll capability.
DC1	Complete device clear capability.
DT0	No device trigger capability.
C0	No controller capability.
E2	Tri-state drivers used on DI0 lines for maximum data transfer rate.

GPIB Connector

An IEEE-488.2 standard GPIB connector is provided for GPIB communication

GPIB Command Format

Most remote commands are the same for both the GPIB and RS-232C ports.

IEEE 488.2 Programming Manual Requirements

This section explains how the ST2400A implements certain standard GPIB functions, as required by IEEE Std. 488.2-1987.

Power-On Settings

When powered-on, the ST2400A will automatically restore all device settings to their last powered settings. No remote commands will affect this power-on rule. However if the instrument detects a problem in non-volatile (battery-backed) RAM at power up, then factory default settings will be restored rather than the last powered settings. A RAM problem at power up will be indicated in the LCD display by the message:

RAM Corruption

Message Exchange

The GPIB message exchange characteristics are as follows:

- The input buffer is command line oriented. A new buffer is provided for each new command line.
- Each command line may contain a maximum of 80 characters.
- The only ST2400A commands that can return more than one message unit are: *lrn?, oh_all?, oh_err_all?
- Command responses are generated immediately. The ST2400A will not wait for a read command before generating a response to a query command.
- No commands are coupled.

IEEE Functional Elements

All functional elements (GPIB Message Types) from the IEEE 488.2-1987 standard that are supported in ST2400A GPIB interface are listed below. For more information refer to your computer or controller documentation and the
IEEE 488.2-1987 standard: Sections 4.3; 7.1.1; 7.3.3 and tables 4.2; 4.3.

1. <PROGRAM MESSAGE>
2. <PROGRAM MESSAGE TERMINATOR>
3. <PROGRAM MESSAGE UNIT>
4. <PROGRAM MESSAGE UNIT SEPARATOR>
5. <COMMAND MESSAGE UNIT>
6. <QUERY MESSAGE UNIT>
7. <COMMAND PROGRAM HEADER> (see note)
8. <QUERY PROGRAM HEADER> (see note)
9. <PROGRAM HEADER SEPARATOR>
10. <PROGRAM DATA SEPARATOR>
11. <PROGRAM DATA>
12. <DECIMAL NUMERIC PROGRAM DATA>
13. <CHARACTER PROGRAM DATA>
14. <NON-DECIMAL NUMERIC PROGRAM DATA>

NOTE: The ST2400A cannot process a <COMPOUND COMMAND PROGRAM HEADER> or a <COMPOUND QUERY PROGRAM HEADER>.

Specific Command Implementations

Reset Command, as defined in the IEEE 488.2-1987 standard, the ST2400A reset command (*rst) does the following:

- Resets all device settings to their default values, except remote interface settings.
- Ignores macros (macros are not implemented in the ST2400A).
- Forces the ST2400A into the Operation Complete Command Idle State (OCIS) and Operation Complete Query Idle State (OQIS).

Overlapped and Sequential Commands: All ST2400A commands are sequential.

Operation Complete Message: ST2400A command operation is always immediate.

GPIB Status and Event Reporting System

The ST2400A GPIB status and event reporting functions are compatible with the IEEE 488.2-1987 standard. The status and event reporting system can be configured to alert the GPIB controller whenever a status change or event occurs. This is accomplished by maintaining two status registers and their associated enable register. The six status registers are:

- Standard Event Status Register (SESR) general purpose status register
- Event Status Enable Register (ESER) enable register
- Status Byte Register (SBR) general purpose status register
- Service Request Enable Register (SRER) enable register
- Test Status Event Register (TSER) enable register
- Test Status Register (TSR) status register

The GPIB bus SRQ (service request) line is asserted when one or more status bits are set. The controller uses a serial poll procedure to find out which instrument initiated the SRQ. Specific status bits can be enable or disable to control which status changes or events result in an SRQ. The ST2400A status registers can be read and set from the RS-232 control interface, however, the SRQ function is only available on the GPIB bus.

SRQs can be generated (or masked off under user control) in response to the ST2400A detecting an alarm or error (LOS. LOF, OFF, REI/FEBE, RDI/FERF. AIS, B1, B2), or in response to an end-of-test (manual stop, or end-of-timed-test).

A status register may contain defined (used) bits and undefined (unused) bits. Each defined bit in a status register corresponds to a given instrument status or event. Once cleared (set to zero), the value of a status bit will remain zero (false) as long as the corresponding event or status does not occur. However, if it is enabled, the value of a status bit will be set to one (true) when the corresponding event or status occurs. Status bits are latched once a status bit is set to one and will remain in this state even when the corresponding event or status becomes false, until the bit is reset by a command or when the ST2400A is powered off.

Each status register has an associated enable register. Moreover, each defined bit in a status register has a corresponding defined bit in the associated enable register. When an enable bit is set to zero, then the corresponding status bit is ignored by the instrument regardless of the state of the relevant event or instrument status. However when an enable bit is set to one, the corresponding status bit is enabled.

GPIB Commands

The contents of any ST2400A status or enable register can be read by an external controller using the appropriate query command. The contents of any enable register can be set to a particular value using the appropriate set command. The contents of the SESR register or the TSR register can be cleared using the Clear Status (*CLS) command. The SESR and the TSR are automatically cleared (reset) after being queried.

The correspondence between ST2400A status registers, enable registers, and commands is summarized below.

Status Registers, Enable Registers, and Related Commands

Status Register	Status Register Query Commands	Enable Register	Enable Register Commands	
			Set	Query
Standard Event Status Register (SESR)	*ESR?	Event Status Enable Register (ESER)	*ESE	*ESE?
Status Byte Register (SBR)	*STB?	Service Request Enable Register (SRER)	*SRE	*SRE?
Test Status Register (TSR)	TSR?	Test Status Enable Register (TSER)	TSE	TSE?

- The *CLS command to clear (reset) the SES, the TSR and return the instrument to the Operation Complete Command Idle State and the Operation Complete Query Idle State. In this state the instrument has no data in its input or output buffers and is not executing a command.
- The *ESR? query command also clears (resets to zero) the SESR status register.
- The TSR? query command also clears (resets to zero) the TSR status register.

Standard Event Status Register

The Standard Event Status Register (SESR) is a status register that indicates errors in command syntax and related occurrences. The name of each bit in the SESR is shown below.

Standard Event Status Register (SESR)

7 PON	6 URQ	5 CME	4 EXE	3 DDE	2 QYE	1 RQC	0 OPC

The function of each bit in the SESR is defined below.

SESR Bit Definitions

Bit	**Definition**
7 (MSB)	PON (Power On): Set to one when the instrument is first powered on.
6	URQ (User Request): Not used.
5	CME (Command Error): Set to one when a command header (mnemonic) has been received that is not recognized or is invalid.
4	EXE (Execution Error): Set to one when an error has occurred while the instrument was responding to a set command or query. May indicate: a) parameter is out of range, b) command uses too many or too few parameters, or c) command cannot be properly executed due to the instrument's current state.
3	DDE (Device Dependent Error): Set to one if input data has been lost at the interface (corrupted characters) or the input buffer has overflowed because a command exceeded the 80 character limit without a terminator.
2	QYE (Query Error): Set to one if: a) an attempt has been made to query the output queue when there are no messages available or pending, b) a command is received but the output queue is not empty, or c) an attempt has been made to query the output queue when an unterminated command is in the input queue.
1	RQC (Request Control): Not used.
0 (LSB)	OPC (Operation Complete): Set to one after a *OPC command is received.

Event Status Enable Register

The Event Status Enable Register (ESER) is the enable register associated with the SESR status register. Each enable bit in the ESER occupies the same bit position as the corresponding status bit in the SESR.

Test Status Register

The Test Status Register (TSR) is a status register that indicates (errors/alarms and end of test conditions). The name of each bit in the TSR is shown below.

Test Status Register (TSR)

7 ---	6 ---	5 ---	4 ERR	3 TRIG	2 SCAN	1 AUTO	0 EOT

The function of each bit in the TSR is defined below.

TSR Bit Definitions

Bit	**Definition**
7 (MSB)	Not defined
6	Not defined
5	Not defined
4	ERR (Error): Set to one when any error or alarm has occurred. **(LOS, LOF, OOF, B1, B2, REI/FEBE, AIS or RDI/FERF)**
3	TRIG: Not used.
2	SCAN: Not used.
1	AUTO: Not used.
0 (LSB)	EOT (End of Test): Set to one on an End of Test condition. **(testing stopped manually, timed test complete, etc.…)**

Test Status Enable Register

The Test Status Enable Register (TSER) is the enable register associated with the TSR status register. Each enable bit in the TSER occupies the same bit position as the corresponding status bit in the TSR. When ERR or EOT are enabled (set to 1), detection of the corresponding test event will cause TSB to be set in the Status Byte Register (SBR).

Status Byte Register

The Status Byte Register (SBR) indicates the overall status of the instrument and is used to control the SRQ line on the GPIB bus. The name of each defined bit in the SBR is shown below.

Status Byte Register (SBR)

7	6	5	4	3	2	1	0
--	MSS	ESB	MAV	TSB	--	--	--

The function of each defined bit in the SBR is described below.

Status Byte Register Bit Definitions

Bit	Definition
7 (MSB)	Not defined.
6	MSS *(Master Status Summary): This bit summarizes the state of the three other bits in the SBR. It is set to 0 if all other bits are 0. It will be set to 1 when any other bit in the SBR equals 1 and its corresponding bit in the SRER is set to one. The status of the MSS bit is reevaluated each time a bit in the SBR or the Service Request Enable Register (SRER) changes.
5	ESB (Event Status Bit): This bit summarizes the Standard Event Status Register (SESR). It is set to 0 if all bits in the SESR are 0. It is set to 1 if any enabled bit in the SESR equals 1. If the ESB is enabled and set to 1, then the RQS and MSS bits will be set to 1. The status of the ESB bit is reevaluated each time one of the bits in the SESR or the Event Status Enable Register (ESER) changes.
4	MAV (Message Available): This bit is set to 1 when there is an output available for the controller.
3	TSB (Test Status Bit): This bit summaries the Test Status Register (TSR). It is set to 0 if all bits in the TSR are 0. It is set to 1 if any enabled bit in the TSR equals 1. If the TSB is enabled and set to 1 in SRER, then the RQS and MSS bits will be set to 1. The status of the TSB bit is reevaluated each time one of the bits in the TSR or the Test Status Enable Register (TSER) changes.
0 - 2	Not defined.

NOTE

*When you use a serial poll to read the SBR, bit 6 is the RQS bit. When you use the *STB? query to read the SBR, bit 6 is the MSS bit.

Service Request Enable Register

The Service Request Enable Register (SRER) is the enable register associated with the SBR status register defined above. The SRER contains an enable bit for the ESB, MAV and TSB bits in the SBR. However there is no enable bit for the MSS or RQS bits which are always enabled.

Remote Commands

Remote Commands, SDH/SONET

This chapter explains the general syntax of the ST2400A remote command language and defines all commands. Command definitions are grouped by function. In addition, an alphabetical list of all ST2400A commands is provided with page references to individual command descriptions.

When using remote commands to set up mapping and channels, it is possible to select an invalid mapping or channel. Care must be taken to match the receiver mappings and channels to the ones available from the ST2400A receiver.

If a GPIB command is entered, but the appropriate hardware for that command is not present in the ST2400A test set, the command will not execute. Certain commands are invalid in Transmitter-only or Receiver-only configurations.

Command Types and Syntax

The ST2400A command set includes two basic types, set and query. The set commands are used to change current status, values, or states—to start or stop a test, for example. While query commands ask the instrument to respond with the contents of a status register, the value of a given setup parameter, a measurement result, or a current state.

Remote commands are directly related to setup parameters and results rather than to front panel keys or setup menus. For example, there is no remote command used to press the SETUP Key.

ST2400A commands may only have a query form or only a set form, however, many have both forms. When a command has both a set and query form, the clear-text header used for the query form will be identical to its corresponding set command form, except for an added (?) question mark. The following examples use clear-text for the view angle query and set commands:

view_angle?	(query command)
view_angle n	(set command)

The view angle query command uses the header **view_angle** with a question mark and no space. The view angle set command uses the same header with a the veritable n (which represents an integer from 0 to 7) with a space.

Command Symbols and Delimiters

<table>
<tr><th>Symbol</th><th>Meaning</th><th></th></tr>
<tr><td>&lt;CR&gt;</td><td>Carriage return (ASCII decimal 13).</td><td></td></tr>
<tr><td>&lt;LF&gt;</td><td>Line Feed (ASCII decimal 10).</td><td></td></tr>
<tr><td>numbers and integers</td><td>Decimal</td><td>10</td></tr>
<tr><td></td><td>Hexadecima</td><td>#HFA</td></tr>
<tr><td></td><td>l:</td><td>#Q377</td></tr>
<tr><td></td><td>Octal:</td><td>#B11111111</td></tr>
<tr><td></td><td>Binary:</td><td></td></tr>
<tr><td></td><td>NOTE: Unless otherwise indicated, all numbers and integers are decimal.</td><td></td></tr>
<tr><td>&lt;string&gt;</td><td>A character string.</td><td></td></tr>
<tr><td>[ ]</td><td>Enclosed argument is required.</td><td></td></tr>
<tr><td>|</td><td>Exclusive OR argument [a | b | c] that means include one and only one of the following parameters: a, b, or c.</td><td></td></tr>
</table>

Multiple Commands

A given command line may contain one or more commands, up to 80 ASCII characters, both upper and lower case. Multiple commands use a semicolon as a delimiter, as follows:

Prompt> header on; sts_oof?; sect_los?; sect_lof?; b1_ber?

Arguments

Commands may include one or more arguments (parameters) following the header. The first argument following the header must be separated from the header by one or more blank characters (spaces). Subsequent arguments must be separated from previous parameters using commas.

Blank Characters (Spaces)

One or more spaces are required between the command header and first parameter (if any). Otherwise spaces are ignored and may be used on a command line between headers, parameters, or required separators for readability.

Arguments Outside of Legal Ranges

If the instrument receives a set command with an argument that is outside the legal range for that argument, then the instrument will set the indicated parameter to its maximum or minimum legal value, depending on which is closer to the received value.

Command Line Terminator

Command lines must be terminated as follows:

- **RS-232 Interface**: Command lines issued to the instrument should be terminated by a simple carriage return (CR). Responses generated by the instrument will be terminated as specified in the RS-232 End-of-Line setup menu for carriage return and line feed functions (CR, LF, CR/LF or LF/CR).
- **GPIB Interface**: Command lines issued to the instrument can be terminated by either EOI (End-Or-Identify) or EOI/LF. Responses generated by the instrument will be terminated as specified in the GPIB setup menu for End-Or-Identify and Line Feed functions (EOI or EOI/LF).

Command Examples

The following is a brief example of an interactive remote session with the command that does the following:

1. Generates a random bit error rate of 1 x 10^{-6}.
2. Enables the Loss Of Signal (LOS) to occur for 135 micro seconds.
3. Sets the Line AIS Indication Signal to “Burst” and enables that Alarm output.
4. Sets the SPE Pointer set during the LOP test to “burst,” with the Pointer value set to 522.

Example interactive session:

parity_rate 6	(sets error generator rate 1 x10^{-6}).
parity_rate?	(queries error generator rate setup)
PARITY_RATE 6	(response)
error_rate on	(turns error generator on)
error_rate?	(queries state of error generator)
ERROR_RATE ON	(response)
los_time 135e-06	(sets LOS generator time to 135 µs)
los_enab	(causes one LOS event)
alm_burst l_ais	(configures alarm burst generator for AIS)
alm_count 100	(sets alarm burst length to 100 STS-n frames)
alm_enab_burst	(generates one AIS alarm burst)

Table Of Commands

Standard Commands

*cls Clear Status

Clears the Standards Events Status Register (SESR) and the Event Status Bit (ESB) in the Status Byte Register (SBR); initiates the Operation Complete Idle State and Operation Complete Query Idle State

Example:	*cls
Response:	NONE
Note:	after execution the ST2400A will have no data in its input and output registers, and will not be executing any command.

*ese[n] Event Status Enable

Sets the Event Status Enable Register (ESER) to the argument n

Argument	n = a decimal number between 0 and 255
Example:	*ese 255
Response:	NONE
Note:	decimal 255 sets the register to 11111111 binary

*ese? Event Status Enable Register Query

Queries the Event Status Enable Register (ESER)

Example:	*ese?
Response:	*ESER [n] (n = a decimal number between 0 and 255)
Note:	decimal 255 sets the register to 11111111 binary

*esr? Event Status Register Query

Returns the content of the Event Status Register (ESR)

Example:	*esr?
Response:	*ESR [n] (n = a decimal number between 0 and 255)
Note:	decimal 255 indicates the register is set to 11111111 binary

*idn? Identify

Returns a character string that includes: company name, instrument model number, and current software version.

Example:	*idn?
Response:	**** TEKTRONIX / MWL, ST2400A, 0, v.vv
Note:	v.vv will be replaced by the software version number

*lrn? Learn Query

List the ST2400A current setup, except for: RS-232C port information.

Example: lrn?

Response: AUTO_VOL 0;
PRINT_PORT PARALLEL;
VIEW_ANGLE 1;
....

Note: The response is a series of commands (headers and parameters) separated by semicolons. They can be stored in the controller and used to restore the same setup later.

*opc? Operation Complete Query

Returns the ASCII character "1" when all previous commands and queries have been completed.

Example: *opc?

Response: *OPC? 1

Note: Used to determine when a group of commands have been completed.

*rst Reset

Returns the ST2400A to its factory default settings and initiates the Operation Complete Idle State and Operation Complete Query Idle State.

Example: *rst

Response: NONE

Note: This command does NOT: change the setup of the RS232C or GPIB ports, alter calibration data, alter the SESR (Standard Event Status Register), alter the ESER (Event Status Enable Register), change power-on status clear flag setting, alter stored word patterns.

*sre [n] Service Request Enable

Sets the content of the Service Request Enable Register (SRER) to value n

Example: *sre 51

Response: NONE

Note: decimal 51 indicates the register is set to 00110011 binary. Using the IEEE 488.2 GPIB standard, register bits are numbered 7 to 0 (left to right). However, the SONET standard labels bits 1 to 8 (left to right). In either case, the MSB is the left most binary bit, and the LSB is the right most binary bit.

*sre? Service Request Enable Query

Returns the content of the Service Request Enable Register (SRER)

Example:	*sre?
Response:	*SRE [n] (n = a decimal number between 0 and 255)
Note:	decimal 255 indicates the register is set to 11111111 binary

*stb? Status Byte Query

Returns the content of the Status Byte Register (SBR)

Example:	*stb?
Response:	*STB [n] (n = a decimal number between 0 and 255)
Note:	decimal 255 indicates the register is set to 11111111 binary

*tst? Self Test

Initiates a self-test and returns the results.

Example:	*tst?
Response:	*TST 0
Note:	A result of 0 (zero) indicates a successful completion of self-test. Any other result would indicate a failure.

*wai Wait

Stop processing any additional commands until all pending operations are completed.

Example:	*wai
Response:	NONE

audio_vol [n] Audio Volume Set command

Sets the audio volume of the internal speaker.

Example:	audio_vol n (n = a decimal integer from 0 to 9)
Response:	NONE
Note:	0 = off, 1 = minimum, and 9 = maximum volume

audio_vol? Audio Volume Query

Returns a decimal integer from 0 to 9, reflecting current audio volume.

Example:	audio_vol?
Response:	AUDIO_VOL n
Note:	n = a decimal integer from 0 to 9

edit_setup [n], "[s]" Edit Setup Label Set

Assigns a label [s] to the saved setup [n]

Augment: n: is the setup number from 0 to 9 in decimal.
s: is the string of

Example: edit_setup 7, "OC-12 Test"

Response: NONE

Note: Up to 10 ASCII characters can be saved with each of the 10 setups.
The string must be enclosed in quotation marks.
Only upper case ASCII characters A to Z and numbers 0 to 9 can be used, along with the following ASCII symbols: # $ % & ' ( ) * + - , / :

edit_setup? [n] Edit Setup Label Query

Returns the current label assigned to the [n] setup saved in memory

Example: edit_setup? [n] (n = a decimal integer between 0 and 9)

Response: EDIT_SETUP OC-12 TEST

Note: Up to 10 ASCII characters can be saved with each of the 10 setups.

gpib_address? GPIB Address Query

Returns the current address value.

Example: gpib_address?

Response: GPIB_ADDRESS 15

Note: This command is only used through the RS-232C Port connection.

gpib_address [n] GPIB Address Set

Sets the value of the GPIB Port address.

Example: gpib_address 15

Response: NONE

Arguments: n = a decimal integer from 0 to 31

Note: This command is only used through the RS-232C Port connection.

gpib_bus [talk_listen | off_bus]   GPIB Bus Mode Set

Sets the GPIB bus mode to on or off.

Argument s:	talk_listen enables the gpib bus.
	off_bus disables the gpib bus. In this mode the gpib bus will not respond to any commands except the GPIB Bus talk_listen from the RS-232C port.
Example:	GPIB Bus off_bus
Response:	NONE
Note:	This command is only used through the RS-232C Port connection. See the gpib_bus? (GPIB Bus Mode Query Command)

gpib_bus?   GPIB Bus Mode Query

Returns the current status of the GPIB bus.

Example:	gpib_bus?
Response:	GPIB_BUS TALK_LISTEN
Note:	The GPIB bus can be off (off_bus) or on (talk_listen) This command is only used through the RS-232C Port connection.

header [on | off]   Header mode command

Enables or disables headers. The header is the part of a response that replicates the command .

Argument s:	on enables headers in the response to any query command.
	off disables headers.
Example:	header on
Response:	NONE
Note:	When the header is on, the query command gpib_bus? will respond with both the header and the status: GPIB_BUS TALK_LISTEN When the header is off, the query command gpib_bus? will respond with the status only: TALK_LISTEN

header?   Header Mode Query

Returns a response string indicating the headers are enabled or disabled.

Example:	header?
Response:	HEADER ON
Note:	See the Header Mode Set Command.

logo? Logo Query command

Returns a character string that includes: company name, instrument model number, current software version, and software date.

Example:	logo?
Response:	**** TEKTRONIX MICROWAVE LOGIC, ST2400A, v.vv yyyy/mm/dd
Note:	v.vv will be replaced by the software version number yyyy/mm/dd will be replaced by the software date

options? Options Query command

Returns one or more character strings indicating which options are installed.

Example:	options?
Response:	OPTIONS RX 1310 RX 1550 TX 1310
Note:	The response indicates that the receiver can receive both the 1310 nm and 1550 nm wavelength, and that the 1310 nm transmitter is installed.

print_port [parallel | serial] Print Port Select

Sets the default printer port to parallel or serial.

Example:	print_port parallel
Response:	NONE
Note:	Sets the default printer port can be set to parallel or serial.

print_port? Print Port Query

Returns the current printer port setting.

Example:	print_port?
Response:	PRINT_PORT PARALLEL
Note:	The default printer port can be set to parallel or serial.

print_string "s" Print String command

Prints the string "s" out the current printer port.

Example:	print_string "This is a test."
Response:	This is a test.
Note:	The character string [s] can be up to 80 (eighty) characters long. It begins and ends with quotation marks.

rs_echo [on | off] RS-232C Echo Control

Enables or disables the echo on the RS-232C port.

Argument s: When the echo is on the ST2400A will echo back each ASCII character sent to the RS-232C port (Full Duplex mode).
When the echo is off the ST2400A will NOT echo back ASCII characters sent to the RS-232C port (Half Duplex mode).

Example: rs_echo on

Response: NONE

rs_echo? RS-232 Echo Control

Returns a character string that indicates whether the echo control is enabled or disabled.

Example: rs_echo?

Response: RS_ECHO ON

Note: Indicates that the echo control is enable (Full Duplex Mode). The response OFF would indicate that the echo control was disable (Half Duplex Mode).

rs_pmt_lf [on | off] RS-232C Prompt Linefeed Set

Enables or disables the current RS-232C EOL (end-of-line) terminator.

Example: rs_pmt_lf on

Response: NONE

Note: The current EOL can be set to CR, LF, CR+LF, or LF+CR.

rs_pmt_lf? RS-232 Prompt Linefeed Query

Returns a character string that indicates the current RS-232C EOL (end-of-line) terminator status.

Example: rs_pmt_lf?

Response: RS_PMT_LF ON

Note: The current EOL can be on (enabled) or off (disabled). Its value is unknown. It may be set to CR, LF, CR+LF, or LF+CR using the quick setup menu.

rs_prompt "s" RS-232 prompt command

Set the character string "s" on the RS-232C prompt.

Example: rs_prompt " ST2400A Unit A> "

Response: ST2400A Unit A>

Note: The character string "s" will be at the start of each new line. Quotation marks must be used before and after the character string, and the string is limited to 12 characters in length.

rs_xon_xoff [on | off] RS-232C Flow Control Set

Enables or disables the RS-232C flow control mode

Example: rs_xon_xoff on
Response: NONE

rs_xon_xoff? RS-232C Flow Control Query

Returns a character string that indicates the current RS-232C flow control status.

Example: rs_xon_xoff?
Response: RS_XON_XOFF ON
Note: The current XON XOFF flow control can be on (enabled) or off (disabled).

General Setup Commands

date “[yy(yy)/mm/dd]” Date Set

Sets the instrument’s internal calendar.

Example:	date “1996/08/30”
Response:	NONE

date? Date Query

Returns a character string indicating the internally stored date.

Example:	date?
Response:	DATE “1996/08/30”

quick_setup [n] Quick Setup (Only valid in Transceivers)

Used to recall one of the four factory default setups:

Argument s:	0 sets the (TX) transmitter to internal and the (RX) receiver to 155Mb/s drop 1 sets the (TX) transmitter to internal and the (RX) receiver to 622Mb/s drop 2 sets the (TX) transmitter to external 155Mb/s and the RX to 155Mb/s drop 3 sets the (TX) transmitter to external 622Mb/s and the RX to 622Mb/s drop
Example:	quick_setup 1
Response:	NONE
Note:	The quick setup command will NOT enable the STM-12 transmit laser.

save_setup [n] Save Setup

Saves the current instrument setup in location n.

Example:	save_setup n (n = a decimal number from 0 to 9)
Response:	NONE

recall_setup [n] Recall Setup

Recalls the saved setup from location n.

Example:	recall_setup n (n = a decimal number from 0 to 9)
Response:	NONE

stored? Stored Setup Query

Returns the currently stored setup locations saved in memory.

Example: stored?
Response: STORED 0,1,2...
Note: Stored locations are numbered from 0 to 9 in decimal.

time? Internal Time Clock Query

Returns the time from the ST2400A internal time of day clock.

Example: time?
Response: TIME "15:30:45"
Note: Uses 24 hours format, HH:MM:SS (Hours Minutes Seconds)

time “[hh:mm:ss]” Internal Time Clock Set

Sets the internally stored time of day clock.

Example: time "15:30:45"
Response: NONE
:
Note: Uses 24 hours format, HH:MM:SS (Hours Minutes Seconds)

tse Test Status Enable Register Set

Sets the content of the TSER (Test Status Enable Register) to a decimal number from 0 to 255

Example: tse 255
Response: NONE
Note: decimal 255 indicates the register is set to 11111111 binary

tse? Test Status Enable Register Query

Returns the content of the TSER (Test Status Event Register)

Example: tse?
Response: TSE [n] (n = a decimal number between 0 and 255)
Note: decimal 255 indicates the register is set to 11111111 binary

tsr? Test Status Register Query

Returns the content of the TSR (Test Status Register).

Example: tsr?
Response: TSR [n] (n = a decimal number between 0 and 255)
Note: decimal 255 indicates the register is set to 11111111 binary

view_angle? Front-panel View Angle Query

Returns a decimal integer that indicates the display angle of view.

Example:	view_angle?
Response:	VIEW_ANGLE 2
Note:	View angle is an integer from 0 to 7. Zero should be the best view angle when the ST2400A is positioned below the operator.

view_angle [n] Front-panel View Angle Set

Sets the view angle to an integer from 0 to 7 decimal.

Example:	view_angle 1
Response:	NONE
Note:	View angle is an integer from 0 to 7. Zero should be the best view angle when the ST2400A is positioned below the operator.

Test Cycle Control Commands

log_print Print Error Log Execute

Prints all error log reports stored in the non-volatile memory Error Log.

Example:	log_print
Response:	NONE

elap_time? Elapsed Test Cycle Time Query

Returns the number of elapsed seconds in the current Test Cycle.

Example:	elap_time?
Response:	ELAP_TIME 120
Note:	Returns the total duration of the last completed Test Cycle, when no Test Cycle is currently running (between Test Cycles).

test_mode [untimed | timed | repeat] Test Mode Set

Sets the test measurement mode.

Argument s	untimed: sets the continuous measurement mode.
	timed: sets the timed measurement mode.
	repeat: sets the repeat-timed measurement mode.
Example:	test_mode timed
Response:	NONE
Note:	See the test time (test_time) set command
	The timed test runs for the duration of time set by the test_time command
	The repeat test runs for the duration of time set by the test_time command

test_mode? Test Mode Query

Returns the current status of the test measurement mode.

Example:	test_mode?
Response:	TEST_MODE TIMED
Note:	See the Test Mode Set Command.

test_print [log | prev | current | off]   Test Print Set

Used to print or stop printing the test results.

Argument s	log: prints the content of the error log.
	prev: prints an end-of-test report at the end of the previous test interval.
	current: prints an end-of-test report from the current test interval.
	off: disables report printing.
Example:	test_print off
Response:	NONE

test_report [log | eot | both | off]   Test Report Set

Enable or disable Test Report Type.

Argument s	log: enable event log reports
	eot: enable end-of-test reports.
	both: enable both log and eot reports.
	off: disables reports.
Example:	test_report both
Response:	NONE

test_report?   Test Report Query

Returns a character string indicating the report status.

Example:	test_report?
Response:	TEST_REPORT BOTH
Note:	See the Test Report Set Command

test_squelch [on | off]   Test Squelch Set

Enables or disables the Squelch.

Argument s	on: (1) logs 10 consecutive events (2) waits for 10 event free seconds.
	off: logs every event.
Example:	test_squelch on
Response:	NONE
Note:	When the Squelch is enabled error logging is limited to ten consecutive reports. Then Error logging is resumed after 10 event free seconds.

test_squelch? Test Squelch Query

Returns the current status of the Squelch

Example:	test_squelch?
Response:	TEST_SQUELCH ON
Note:	See the Test Squelch Set Command.

test_state [run | stop] Test State Set

Sets the test state to either run or stop.

Arguments	run: clears all counters, clears the history, and starts the test cycle. stop: suspends the test cycle.
Example:	test_state run
Response:	NONE
Note:	See pause_state command. The Trouble Scan Mode must be OFF before the Test State can be enabled.

test_state? Test State Query

Returns the current status of the test cycle.

Example:	test_state?
Response:	TEST_STATE ON
Note:	The test cycle can be either on (running) or off. When the test state = stop the pause-run-resume sequence can be used to save the current data and history.

test_time “[d-hh:mm:ss]” Test Time Set

Sets the duration of the test cycle in days (d), hours, (hh) minutes (mm), and seconds (ss).

Example:	test_time “0-02:00:00”
Response:	NONE
Note:	The Argument must be enclosed in quotation marks. One hyphen and two colons must be used as shown in the example. Maximum Test Time = 9 days, 23 hours, 59 minutes, 59 seconds.

test_time? Test State Query

Returns a character string indicating the current test time settings.

Example:	test_time?
Response:	TEST_TIME “0-02:00:00”

test_state [run | stop]   Test State Set

Sets the test state to either run or stop.

Arguments run clears all counters, clears the history, and starts the test cycle.
off stops the test cycle.
Example: test_state run
Response: NONE
Note: See pause_state command

test_state?   Test State Query

Returns the current status of the test cycle.

Example: test_state?
Response: TEST_STATE ON
Note: The test cycle can be either on (running) or off.

pause_state [pause | resume]   Pause State Set

Sets the test state to either pause or resume.

Arguments pause interrupts the test cycle.
resume restarts the test cycle.
Example: pause_state pause
Response: NONE
Note: See test_state command

pause_state?   Pause State Query

Returns the current status of the test cycle.

Example: pause_state?
Response: PAUSE_STATE PAUSE
Note: The test cycle state can be paused (interrupted) or resume (running).

System Configuration

m_mode? Measurement Mode Query

Returns a character string indicating the currently active measurement mode.

Example: m_mode?
Response M_MODE SDH
Note: The current measure mode can be either SDH or SONET

Command	**Result**	**Available Options**
Remote changing of error menus menu **meas_mode,menu_type** example: ***menu*** *SDH_ERRORS,B1_ERR*	will switch to menu specified	Available *meas_mode* options: SONET_ERRORS SDH_ERRORS Available *menu_type* for **SONET**: B1_ERR, B2_ERR, B3_ERR, PB_ERR, TOTAL, TOTAL2, TIME Avaialable *menu_type* for **SDH**: B1_ERR,B1_ERR2,B1_ERR3, B1_ERR4,B1_ERR5,B2_ERR, B2_ERR2,B2_ERR3,B2_ERR4, B2_ERR5,B3_ERR,B3_ERR2,B3_ERR3 ,B3_ERR4,B3_ERR5, REI_ERR, REI_ERR2,REI_ERR3,REI_ERR4,REI _ERR5, HP_ERR, HP_ERR2, HP_ERR3,HP_ERR4,HP_ERR5, TSE_ERR, TSE_ERR2, TSE_ERR3, TSE_ERR4,TSE_ERR5, TOTAL,TOTAL2, TIME

new_m_mode [sdh | sonet] New Measurement Mode Set

Sets the measurement mode to be used after the next power-on cycle.

Example: new_m_mode sonet
Response NONE
Note: see (new_m_mode?) New Measurement Mode Query Command

new_m_mode? New Measurement Mode Query

Returns a character string indicating the measurement mode that is set to be used after the next power-on cycle.

Example: new_m_mode sonet
Response NONE
Note: see (new_m_mode?) New Measurement Mode Set Command

log_clear Log Clear

Clears all error log entries.

Example: log_clear
Response: NONE

resume_laser [off | on]

Sets the operation of the laser after a power loss

Argument	resume_laser off	Disables the laser on power-up
	resume_laser on	Restores last setting of the RX or TX laser on power-up, and restores any active tests, error or alarm injection.

Example: resume_laser on
Response:

resume_laser?

Queries the current setting. Control the laser after a power loss

Example: resume_laser?
Response:

rx_installed? Receiver Board Query

Returns receiver board status that was detected during the power–on sequence

Example: rx_installed?
Response: RX_INSTALLED YES
Note: YES indicates that the receiver board was detected.
NO indicates that the receiver board was NOT detected.

tx_installed? Transmitter Board Query

Returns transmitter board status that was detected during the power–on sequence

Example: rx_installed?
Response: RX_INSTALLED YES
Note: YES indicates that the transmitter board was detected.
NO indicates that the transmitter board was NOT detected.

tx_wav? TX Wavelength Query

Returns transmitter wavelength

Example: tx_wav?

Response: TX_WAVE 1310
Note: The transmitter wavelength can be switched to either 1310nm or 1550nm in a dual wavelength unit.

tx_wav [1310 | 1550] TX Wavelength Set

Switches the transmitter wavelength in a dual wavelength unit.

Example: tx_wav 1550
Response: NONE
Note: The transmitter wavelength can be switched to either 1310nm or 1550nm.

Receiver Remote Commands

If a GPIB command is entered, but the appropriate hardware for that command is not present in the ST2400A test set, the command will not execute. Certain commands are invalid in Transmitter-only or Receiver-only configurations.

rx_2488_in [opt | elec]

Set the Receiver Input state to Optical ("OPT") or Electrical ("ELEC")

Example:	rx_2488_in opt

rx_2488_in?

Query the "OPT/ELEC" state of the Receiver 2.488 Gb/s input

Example:	rx_2488_in?
Response	OPT

rx_apd_info?

Queries the APD module (if one is installed) and print out a detailed description in the following format:

board_type, rev_level, options, cal_date

Example:	rx_apd_info?
Response	0X30, A1, 1, 03/02/98

rx_elec_io?

Queries the Receiver to determine the status of the electrical input

rx_elec_io?	will query the status of the electrical input signal, either “**OK**” or “**INVALID**” *example return value:* ***INVALID***

rx_opt_alarm [reset]

Reset the APD alarm circuitry

Example:	rx_opt_alarm reset

rx_opt_alarm?

Query the APD alarm status, either "OK" or "ALARM"

Example:	rx_opt_alarm?
Response	ALARM

rx_thresh

Sets the Receiver Threshold offset

Example: rx_thresh -50

rx_thresh?

Queries the Receiver Threshold offset

Example: rx_thresh?

Response -50

RX Mapping

rx_map_chan?

Query - Channel mapping

Example: rx_map_chan?

rx_map_chan [1-48]

Setting Channel mapping

Example: rx_map_chan 16

Note: dependent on rx_mapping that is selected.

rx_mapping?

Query - mapping on RX

Example: rx_mapping?

rx_mapping [sts1 | sts3c | sts12c | sts48c]

SONET- Setting mapping signal

Example: rx_mapping

rx_mapping [stm0 | vc4 | vc4-4c | vc4-16c]

SDH - Setting mapping signal

Example: rx_mapping

rx_test_patt?

Query - Test Pattern - RX

Example: rx_test_patt?

Response

rx_test_patt [pn23 | pn23inv | pn31 | pn31inv | fixed= (00 to FF) |

Setting Test Pattern on RX

Example: rx_test_patt?

Response pn31inv

Note: rx_test_patt fixed=00 is equivalent to rx_test_patt allzeros

Pointers

rx_neg_pj?

Query -- Negative Pointer Justification during test

Example: rx_neg_pj?

Response

Value set to zero everytime a test is started.

rx_pos_pj?

Query -- Positive Pointer Justification during test

Example: rx_pos_pj?

Response

Value set to zero everytime a test is started.

rx_spe_ndf?

Query -- Synchronous Payload Envelope - New Data Flag during test

Example: rx_spe_ndf?

Response

Value set to zero everytime a test is started.

rx_spe_pntr?

Query -- live status of Synchronous Payload Envelope Pointer

Example: rx_spe_pntr?

Response

Value set to zero everytime a test is started.

B1, B2, B3, Payload Bit, Path REI, FEBE TSE Error Data Query Commands

b1_err? B1 Total Error Count

Returns the total number of B1 errors recorded during the test cycle.

Example: b1_err?
Response B1_ERR [b1 error data]
Note: b1 error data includes: total count, error rate (X.Y e-Z), errored seconds, and severely errored seconds.

b2_err? B2 Total Error Count

Returns the total number of B2 errors recorded during the test cycle.

Example: b2_err?
Response B2_ERR [b2 error data]
Note: b2 error data includes: total count, error rate (X.Y e-Z), errored seconds, and severely errored seconds.

b3_err? B3 Total Error Count

Returns the total number of B3 errors recorded during the test cycle.

Example: b3_err?
Response B3_ERR [b3 error data]
Note: B3 error data includes: total count, error rate (X.Y e-Z), errored seconds, and severely errored seconds.

path_rei_err? Path REI Total Error Count (SDH)

Returns the total number of Path REI errors recorded during the test cycle.

Example: path_rei_err?
Response Path_REI_ERR

pb_err? Pattern Bit Total Error Count (SONET)

Returns the total number of Pattern Bit errors recorded during the test cycle.

Example: pb_err?
Response PB_ERR

tse_err? TSE Total Error Count (SDH)

Returns the total number of TSE errors recorded during the test cycle.

Example: tse_err?
Response tse_ERR

sts_febe_err? FEBE Total Error Count (SONET)

Returns the number of path Far End Block Errors counted in the received signal.

Example: sts_febe_err?
Response STS_FEBE_ERR [n] (n = the total count in decimal)
:

b1_ber? B1 Bit Error Rate Query

Returns the number of STS-n section BIP-8 code violation bit error rate measured in the received signal.

Example: b1_ber?
Response B1_BER [NR3 n]
Note: NR3 is the is the error identification and n = the current rate.
The rate is returned as a negative exponent, such as 1.00e-6
1.00e-6 indicates 1 errored bit in each one million bits.

b2_ber? B2 Bit Error Rate Query

Returns the number of STS-n line BIP-8 code violation bit error rate measured in the received signal.

Example: b2_ber?
Response B2_BER [NR3 n]
Note: NR3 is the is the error identification and n = the current rate
The rate is returned as a negative exponent, such as 1.00e-6
1.00e-6 indicates 1 errored bit in each one million bits.

b3_ber? B3 Bit Error Rate Query

Returns the number of STS-n line BIP-8 code violation bit error rate measured in the received signal.

Example: b3_ber?
Response B3_BER [NR3 n]
Note: NR3 is the is the error identification and n = the current rate
The rate is returned as a negative exponent, such as 1.00e-6
1.00e-6 indicates 1 errored bit in each one million bits.

path_rei_ber? Path-REI Bit Error Rate Query (SDH & SONET))

Returns the number of Path REI violation bit error rate measured in the received signal.

Example: path_rei_ber?
Response PATH_REI_BER [NR3 n]
Note: NR3 is the is the error identification and n = the current rate
The rate is returned as a negative exponent, such as 1.00e-6

1.00e-6 indicates 1 errored bit in each one million bits.

pb_ber? Pattern Bit Bit Error Rate Query (SONET)

Returns the number of Pattern Bit violation bit error rate measured in the received signal.

Example:	pb_ber?
Response	PB_BER [NR3 n]
Note:	NR3 is the is the error identification and n = the current rate The rate is returned as a negative exponent, such as 1.00e-6 1.00e-6 indicates 1 errored bit in each one million bits.

tse_ber? TSE Bit Error Rate Query (SDH)

Returns the number of TSE bit error rate measured in the received signal.

Example:	tse_ber?
Response	Tse_BER [NR3 n]
Note:	NR3 is the is the error identification and n = the current rate The rate is returned as a negative exponent, such as 1.00e-6 1.00e-6 indicates 1 errored bit in each one million bits.

RS-B1, MS-B2, and MS-REI Error Data Query Commands

G.826 and M.2101 Measurements in SDH Mode

In SDH mode, the following measurements are calculated for each of the B1, B2, and MS-REI error types:

TOT (Total Errors) Total number of errors detected since testing was started (or restarted).

BER (Bit Error Ratio) Ratio of number of errors detected to the number of bits monitored for errors since testing was started (or restarted). (Computed only for B1, B2 errors; Not computed for MS-REI errors.)

ES (Errored Seconds) Number of seconds since testing was started (or restarted) in which one or more errors were detected.

SES (Severely Errored Seconds) Number of seconds since testing was started (or restarted) in which the number of detected errors exceeded the severely errored second threshold or in which a relevant defect was detected. Relevant defects are: LOS (all SES), LOF (all SES), MS-AIS (B2 and MS-REI SES only), and MS-RDI (MS-REI SES only). SES threshold is currently set to 30% of the blocks (SDH frames) per second for B1, B2, and MS-REI errors. (Thresholds follow G.826 recommendation, but do not conform to draft new recommendation G.epmrs, “Error Performance Events for SDH Multiplex and Regenerator Sections”, which recommend 30% threshold for B1 errors, but 15% threshold for B2 and MS-REI errors.)

CSES (Consecutive Severely Errored Seconds) Count of Consecutive Severely Errored Second periods. A CSES period is defined as a period of more than 2, but less than 10, consecutive SES.

EB (Errored Blocks) The number of SDH frames in which one or more errors of the selected type were detected since testing was started (or restarted). SDH frames occur at a rate of 8000 per second. Per ITU-T G.826, Errored Blocks should count one EB for each frame in which one or more errors are detected, but may count the total number of errors detected per frame (see G.826 C.1.5). The ST-2400 and ST-2400A count EBs as the total number of errors detected.

BBE (Background Block Errors) The number of errored blocks not occurring during SES or Unavailable time.

BBER (Background Block Error Ratio) The ratio of Background Block Errors (BBE) to total blocks in available time since testing was started (or restarted). The count of total blocks excludes all blocks during SESs.

ESR (Errored Second Ratio) The ratio of Errored Seconds (ES) to total seconds in available time since testing was started (or restarted).

SESR (Severely Errored Second Ratio) The ratio of Severely Errored Seconds (SES) to total seconds in available time since testing was started (or restarted).

AEB (Available Errored Blocks) Number of Errored Blocks (EB) counted during available time.

ABBE (Available Background Block Errors) Number of Background Block Errors (BBE) counted during available time.

AES (Available Errored Seconds) Number of Errored Seconds (ES) counted during available time.

ASES (Available Severely Errored Seconds) Number of Severely Errored Seconds (SES) counted during available time.

UAS (Unavailable Seconds) The number of seconds during which error performance was such that the received signal was deemed to be unavailable (see definition of available and unavailable time below).

UR (Unavailable Seconds Ratio) The ratio of unavailable seconds to total time since testing was started (or restarted).

Definition of Available and Unavailable Time (per G.826) Received signal becomes unavailable when 10 consecutive severely errored seconds are detected. The 10 consecutive SES required to enter the unavailable state are considered unavailable seconds. The signal remains in the unavailable state until 10 consecutive non-severely errored seconds are detected. The 10 consecutive non-severely errored seconds are considered part of the available time.

b1_eb? B1 Errored Block Count

Returns the number of corrupted section blocks counted in the received signal.

Example: b1_eb?
Response: B1_EB [c] (c = counted seconds in decimal)
Note: RS-B1 error blocks include the total count, conforms to ITU G.826

b2_eb? B2 Errored Block Count

Returns the number of corrupted line blocks counted in the received signal.

Example: b2_eb?
Response: B2_EB [c] (c = counted seconds in decimal)
Note: MS-B2 error blocks include the total count, conforms to ITU G.826

b3_eb? B3 Errored Block Count

Returns the number of corrupted line blocks counted in the received signal.

Example: b3_eb?
Response: B3_EB [c] (c = counted seconds in decimal)
Note: MS-B2 error blocks include the total count, conforms to ITU G.826

path_rei_eb? Path REI Errored Block Count

Returns the number of path remote error indications counted in the receiver signal.

Example: path_rei_eb?
Response: Path_REI_EB [c] (c = counted seconds in decimal)
Note: MS-REI error blocks include the total count, conforms to ITU G.826

tse_eb? TSE Errored Block Count

Returns the number of TSE error indications counted in the receiver signal.

Example: tse_eb?
Response: Tse_EB [c] (c = counted seconds in decimal)
Note: MS-REI error blocks include the total count, conforms to ITU G.826

b1_bbe? B1 Background Block Error Count

Returns the number of b1 errored blocks counted that occurred outside the SES.

Example: b1_bbe?
Response: B1_BBE [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826. Also see b1_ses?

b2_bbe? B2 Background Block Error Count

Returns the number of b2 errored blocks counted that occurred outside the SES.

Example: b2_bbe [b2 background block errors]
Response: B2_BBE [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826. Also see b2_ses?

b3_bbe? B3 Background Block Error Count

Returns the number of b3 errored blocks counted that occurred outside the SES.

Example: p3_bbe [b2 background block errors]
Response: B3_BBE [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826. Also see b3_ses?

path_rei_bbe? Path REI Background Block Error Count

Returns the number of path REI errored blocks counted that occurred outside the SES.

Example: path_rei_eb?
Response: Path_REI_BBE [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826. Also see path_rei_ses?

tse_bbe? TSE Background Block Error Count

Returns the number of TSE errored blocks counted that occurred outside the SES.

Example: tse_eb?
Response: tse_BBE [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826. Also see tse_ses?

b1_bber? B1 Background Block Error Ratio

Returns the ratio of b1 errored blocks that occurred outside the SES.

Example:	b1_bbe?
Response:	B1_BBE [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826. Also see b1_ses?

b2_bber? B2 Background Block Error Ratio

Returns the ratio of b2 errored blocks that occurred outside the SES.

Example:	b2_bbe [b2 background block errors]
Response:	B2_BBE [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826. Also see b2_ses?

b3_bber? B3 Background Block Error Ratio

Returns the ratio of b32 errored blocks that occurred outside the SES.

Example:	b3_bbe [b2 background block errors]
Response:	B3_BBE [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826. Also see b3_ses?

path_rei_bber? Path REI Background Block Error Ratio

Returns the ratio of path rei errored blocks that occurred outside the SES.

Example:	path_rei_eb?
Response:	Path_REI_BBE [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826. Also see path_rei_ses?

tse_bber? TSE Background Block Error Ratio

Returns the ratio of TSE errored blocks that occurred outside the SES.

Example:	tse_eb?
Response:	tse_BBE [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826. Also see tse_ses?

b1_es? B1 Errored Seconds Query

Returns the number of STM-n section BIP-8 code violation errored seconds counted in the received signal.

Example: b1_es?
Response: B1_ES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826.

b2_es? B2 Errored Seconds Query

Returns the number of STM-n line BIP-24 code violation errored seconds counted in the received signal.

Example: b2_es?
Response: B2_ES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826.

b3_es? B3 Errored Seconds Query

Returns the number of B3errored seconds counted in the received signal.

Example: b3_es?
Response: B3_ES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826.

path_rei_es? Path REI Errored Seconds Query

Returns the number of Path REI errored seconds counted in the received signal.

Example: path_rei_es?
Response: Path_REI_ES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826.

tse_es? TSE Errored Seconds Query

Returns the number of TSE errored seconds counted in the received signal.

Example: tse_es?
Response: tse_ES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.826.

b1_ses? B1 Severely Errored Seconds Query

Returns the number of STM-n section BIP-8 code violation severely errored seconds counted in the received signal.

Example:	b1_ses?
Response:	B1_SES [c] (c = counted seconds in decimal)
Note:	Conforms to ITU G.826.

b2_ses? B2 Severely Errored Seconds Query

Returns the number of STM-n line BIP-24 code violation severely errored seconds counted in the received signal.

Example:	b2_ses?
Response:	B2_SES [c] (c = counted seconds in decimal)
Note:	Conforms to ITU G.826.

b3_ses? B3 Severely Errored Seconds Query

Returns the number of B3 severely errored seconds counted in the received signal.

Example:	b3_ses?
Response:	B3_SES [c] (c = counted seconds in decimal)
Note:	Conforms to ITU G.826.

path_rei_ses? Path REI Severely Errored Seconds Query

Returns the number of Path REI severely errored seconds counted in the received signal.

Example:	path_rei_ses?
Response:	Path_REI_SES [c] (c = counted seconds in decimal)
Note:	Conforms to ITU G.826.

tse_ses? TSE Severely Errored Seconds Query

Returns the number of TSE severely errored seconds counted in the received signal.

Example:	tse_ses?
Response:	tse_SES [c] (c = counted seconds in decimal)
Note:	Conforms to ITU G.826.

b1_cses? B1 Consecutively Severely Error Seconds

Queries the total Consecutively Severely Errored Seconds for B1 errors

Example:	b1_cses?
Return value	0

b2_cses? B2 Consecutively Severely Error Seconds

Queries the total Consecutively Severely Errored Seconds for B2 errors

Example:	b2_cses?
Return value	0

b3_cses? B3 Consecutively Severely Error Seconds

Queries the total Consecutively Severely Errored Seconds for B3 errors

Example:	b3_cses?
Return value	0

path_rei_cses? Path REI Consecutively Severely Error Seconds

Queries the total Consecutively Severely Errored Seconds for Path REI errors

Example:	path_rei_cses?
Return value	0

tse_cses? TSE Consecutively Severely Error Seconds

Queries the total Consecutively Severely Errored Seconds for TSE errors

Example:	tse_cses?
Return value	0

b1_esr? B1 Errored Seconds Ratio Query

Returns the number of STM-n section BIP-8 code violation errored seconds counted in the received signal divided by the available time.

Example: b1_esr?
Response: B1_ESR [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.826.

b2_esr? B2 Errored Seconds Ratio Query

Returns the number of STM-n line BIP-24 code violation errored seconds counted in the received signal divided by the available time.

Example: b2_esr?
Response: B2_ES [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.826.

b3_esr? B3 Errored Seconds Ratio Query

Returns the number of B3 errored seconds counted in the received signal divided by the available time.

Example: b3_esr?
Response: B3_ES [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.826.

path_rei_esr? Path REI Errored Seconds Ratio Query

Returns the number of Path REI errored seconds counted in the received signal.

Example: path_rei_esr?
Response: Path_REI_ES [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.826.

tse_esr? TSE Errored Seconds Ratio Query

Returns the number of TSE errored seconds counted in the received signal.

Example: tse_esr?
Response: Tse_ES [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.826.

b1_sesr? B1 Severely Errored Seconds Ratio Query

Returns the number of STM-n section BIP-8 code violation severely errored seconds counted in the received signal divided by the available time.

Example:	b1_ses?
Response:	B1_SES [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826.

b2_sesr? B2 Severely Errored Seconds Ratio Query

Returns the number of STM-n line BIP-24 code violation severely errored seconds counted in the received signal divided by the available time.

Example:	b2_ses?
Response:	B2_SES [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826.

b3_sesr? B3 Severely Errored Seconds Ratio Query

Returns the number of B3 severely errored seconds counted in the received signal divided by the available time.

Example:	b3_ses?
Response:	B3_SES [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826.

path_rei_sesr? Path REI Severely Errored Seconds Ratio Query

Returns the number of Path REI severely errored seconds counted in the received signal divided by the available time.

Example:	path_rei_ses?
Response:	Path_REI_SES [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826.

TSE_sesr? TSE Severely Errored Seconds Ratio Query

Returns the number of TSE severely errored seconds counted in the received signal divided by the available time.

Example:	Tse_ses?
Response:	Tse_SES [e] (e = expressed as a negative exponent)
Note:	Conforms to ITU G.826.

b1_uas? B1 Unavailable Seconds Query

Returns the number of STM-n section BIP-8 code violation unavailable seconds counted in the received signal.

Example: b1_uas?
Response: B1_UAS [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.827.

b2_uas? B2 Unavailable Seconds Query

Returns the number of STM-n line BIP-24 code violation unavailable seconds counted in the received signal.

Example: b2_uas?
Response: B2_UAS [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.827.

b3_uas? B3 Unavailable Seconds Query

Returns the number of B3 unavailable seconds counted in the received signal.

Example: b3_uas?
Response: B3_UAS [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.827.

path_rei_uas? Path REI Unavailable Seconds Query

Returns the number of Path REI unavailable seconds counted in the received signal.

Example: path_rei_uas?
Response: Path_REI_UAS [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.827.

TSE_uas? TSE Unavailable Seconds Query

Returns the number of TSE unavailable seconds counted in the received signal.

Example: tse_uas?
Response: tse_UAS [c] (c = counted seconds in decimal)
Note: Conforms to ITU G.827.

b1_ur? B1 Unavailable Seconds Ratio Query

Returns the number of STM-n section BIP-8 code violation errored seconds counted in the received signal.

Example: b1_ur?
Response: B1_UR [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.827.

b2_ur? B2 Unavailable Seconds Ratio Query

Returns the number of STM-n line BIP-24 code violation errored seconds counted in the received signal.

Example: b2_ur?
Response: B2_UR [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.827.

b3_ur? B3 Unavailable Seconds Ratio Query

Returns the number of B3 errored seconds counted in the received signal.

Example: b3_ur?
Response: B3_UR [e] (e = expressed as a negative exponent)
Note: Conforms to ITU G.827.

path_rei_ur? Path REI Unavailable Seconds Ratio Query

Returns the number of Path REI unavailable seconds counted in the received signal.

Example: path_rei_ur?
Response: Path_REI_UR [e] (e= expressed as a negative exponent)
Note: Conforms to ITU G.827

TSE_ur? TSE Unavailable Seconds Ratio Query

Returns the number of TSE unavailable seconds counted in the received signal.

Example: tse_ur?
Response: TSE_UR [e] (e= expressed as a negative exponent)
Note: Conforms to ITU G.827

b1_aeb? B1 Available Time Errored Block Count

Returns the number of corrupted section blocks counted during the available time.

Example:	b1_aeb?
Response:	B1_AEB [c] (c = counted blocks, in decimal)
Note:	Conforms to ITU G. 2101

b2_aeb? B2 Available Time Errored Block Count

Returns the number of corrupted line blocks counted during the available time.

Example:	b2_aeb?
Response:	B2_AEB [c] (c = counted blocks, in decimal)
Note:	Conforms to ITU G. 2101

b3_aeb? B3 Available Time Errored Block Count

Returns the number of corrupted line blocks counted during the available time.

Example:	b3_aeb?
Response:	B3_AEB [c] (c = counted blocks, in decimal)
Note:	Conforms to ITU G. 2101

path_rei_aeb? Path REI Available Time Errored Block Count

Returns the number of path remote error indications counted during available time..

Example:	path_rei_aeb?
Response:	Path_REI_AEB [c] (c = counted blocks, in decimal)
Note:	Conforms to ITU G. 2101

tse_aeb? TSE Available Time Errored Block Count

Returns the number of path remote error indications counted during available time..

Example:	tse_aeb?
Response:	tse_AEB [c] (c = counted blocks, in decimal)
Note:	Conforms to ITU G. 2101

b1_abbe? B1 Available Time Background Block Errors

Returns the number of corrupted section background blocks counted during the available time.

Example: b1_abbe?
Response: B1_ABBE [c] (c = counted background blocks, in decimal)
Note: Conforms to ITU G. 2101

b2_abbe? B2 Available Time Background Block Errors

Returns the number of corrupted line background blocks counted during the available time.

Example: b2_abbe?
Response: B2_ABBE [c] (c = counted background blocks, in decimal)
Note: Conforms to ITU G. 2101

b3_abbe? B3 Available Time Background Block Errors

Returns the number of corrupted line background blocks counted during the available time.

Example: b3_abbe?
Response: B3_ABBE [c] (c = counted background blocks, in decimal)
Note: Conforms to ITU G. 2101

path_rei_abbe? Path REI Available Time Background Block Errors

Returns the number of background remote error indications counted during available time..

Example: path_rei_abbe?
Response: Path_REI_ABBE [c] (c = counted background blocks, in decimal)
Note: Conforms to ITU G. 2101

TSE_abbe? TSE Available Time Background Block Errors

Returns the number of background remote error indications counted during available time..

Example: tse_abbe?
Response: tse_ABBE [c] (c = counted background blocks, in decimal)
Note: Conforms to ITU G. 2101

b1_aes? B1 Available Time Errored Seconds Query

Returns the number of STM-n section BIP-8 code violation errored seconds counted in the received signal, during the available time.

Example: b1_aes?
Response: B1_AES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G. 2101

b2_aes? B2 Available Time Errored Seconds Query

Returns the number of STM-n line BIP-24 code violation errored seconds counted in the received signal, during the available time.

Example: b2_aes?
Response: B2_AES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G. 2101

b3_aes? B3 Available Time Errored Seconds Query

Returns the number of B3 errored seconds counted in the received signal, during the available time.

Example: b3_aes?
Response: B3_AES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G. 2101

path_rei_aes? Path REI Available Time Errored Seconds Query

Returns the number of errored seconds counted in the received signal, during the available time.

Example: path_rei_aes?
Response: Path_REI_AES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G. 2101

tse_aes? TSE Available Time Errored Seconds Query

Returns the number of errored seconds counted in the received signal, during the available time.

Example: tse_aes?
Response: tse_AES [c] (c = counted seconds in decimal)
Note: Conforms to ITU G. 2101

b1_ases? B1 Available Time Severely Errored Seconds

Returns the number of STM-n section BIP-8 code violation severely errored seconds counted in the received signal, during the available time.

Example:	b1_ases?
Response:	B1_ASES [c] (c = counted seconds, in decimal)
Note:	Conforms to ITU G. 2101

b2_ases? B2 Available Time Severely Errored Seconds

Returns the number of STM-n line BIP-24 code violation severely errored seconds counted in the received signal, during the available time.

Example:	b2_ases?
Response:	B2_ASES [c] (c = counted seconds, in decimal)
Note:	Conforms to ITU G. 2101

b3_ases? B3 Available Time Severely Errored Seconds

Returns the number of B3 severely errored seconds counted in the received signal, during the available time.

Example:	b3_ases?
Response:	B3_ASES [c] (c = counted seconds, in decimal)
Note:	Conforms to ITU G. 2101

path_rei_ases? Path REI Available Time Severely Errored Seconds

Returns the number of Path REI severely errored seconds counted in the received signal, during the available time.

Example:	path_rei_ases?
Response:	Path_REI_ASES [c] (c = counted seconds, in decimal)
Note:	Conforms to ITU G. 2101

tse_ases? TSE Available Time Severely Errored Seconds

Returns the number of TSE severely errored seconds counted in the received signal, during the available time.

Example:	tse_ases?
Response:	tse_ASES [c] (c = counted seconds, in decimal)
Note:	Conforms to ITU G. 2101

SDH Error Status

hp_b3?

Query - High Path B3 error condition (SDH)

Example hp_b3?

Response

hp_rei?

Query - High Path REI error condition (SDH)

Example hp_rei?

Response

tse_stat?

Query - TSE condition (SDH)

Example tse_stat?

Response

SONET Error Status

path_b3?

Query - Path B3 error condition (SONET)

Example path_b3?

Response

path_rei?

Query - Path REI error condition (SONET)

Example path_rei?

Response

pyld_bit?

Query - Payload Bit error condition (SONET)

Example pyld_bit?

Response

LED Status Query Commands

led_stat? LED Status Query

Returns the current state of all front panel status indicators (LEDs).

Example: led_stat?

Response: LED_STAT? CLEAR

Note: The LED status indicator can be in one of three states:
CLEAR indicates that all of the LED status indicators are off and have been off for the total duration of the current test cycle.
CUR (current) indicates that one or more LED indicators are on or had been on since last queried during the current test cycle.
HIST (history) is only available if querying specific errors. See each following command within this LED Status Query Command section (this page and four following pages).
The led_stat? command will report the CUR condition until the user remotely queries and gets the status information from all of the LED indicators that were active at any time during the test cycle. Each LED status command will also report the CUR condition if the LED was activated since its last query. Once all are queried, each LED indicator and the led_stat? command will report the HIST condition if an error or alarm had ever occurred during the current test cycle, otherwise the CLEAR condition will be reported.

Command	**Result**	**Sample of returned data**
led_sum?	Returns a quick status of all LED status commands, without having to individually query each of them, also gives B1, B2, B3, PB, Path REI error rates and counts, and FEBE counts. B3, Payload Bit (PB), Path REI, LOP, Path AIS, and Path RDI errors are only displayed when B3-capable hardware is present in the ST2400A test unit. Works the same in SONET and SDH modes, only SONET related commands listed. **Related commands:** sect_los?, sect_lof?, sect_oof?, sect_b1?, line_b2?, line_ais?, line_ferf?,line_febe?,b1_err?, b2_err?,b3_err?, b1_ber?,b2_ber?, b3_ber?, pb_err?, path_rei_err? sts_febe_err?, path_lop?, path_ais,	LOS = HIST LOF = HIST OOF = HIST B1 ERR = HIST B2 ERR = HIST B3 ERR = HIST REI/FEBE = HIST RDI/FERF = HIST Path AIS = HIST Path RDI = HIST B1 COUNT = 133384 B1 BER = 7.7E-07 B2 COUNT = 3105246 B2 BER = 1.8E-05 B3 COUNT = 133384 B3 BER = 7.7E-07 PB COUNT = 133384 PB BER = 7.7E-07 REI/FEBE COUNT = 1750132 Path REI COUNT = 1750132

path_rei?, path_rdi

sect_los? Section Loss Signal Query (SONET)

Request the current state of the LOS indicator.

Example: sect_los?

Response: SECT_LOS? CLEAR

Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the LOS LED is now off and had been off during the current test cycle.
CUR (current) indicates that the LOS indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the LOS LED is off, but had been on during some period before the last query, during the current test cycle.

sect_lof? Section Loss Frame Query (SONET)

Request the current state of the LOF indicator.

Example: sect_lof?

Response
: SECT_LOF? CLEAR

Note: The section LOF indicator can be in one of three states:
CLEAR indicates that the LOF LED is now off and had been off during the current test cycle.
CUR (current) indicates that the LOF indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the LOF LED is off, but had been on during some period before the last query, during the current test cycle.

sts_oof? STSn Out-Of-Frame Query (SONET)

Request the current state of the OOF indicator.

Example: sts_oof?

Response
: STS_OOF? CLEAR

Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the OOF LED is now off and had been off during the current test cycle.
CUR (current) indicates that the OOF indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the OOF LED is off, but had been on during some period before the last query, during the current test cycle.

sect_b1? Section B1 Query (SONET)

Request the current state of the section B1 byte indicator.

Example: sect_b1?

Response
: SECT_B1? CLEAR

Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the B1 LED is now off and had been off during the current test cycle.
CUR (current) indicates that the B1 indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the B1 LED is off, but had been on during some period before the last query, during the current test cycle.

line_b2? Line B2 Query (SONET)

Request the current state of the line B2 byte indicator LED.

Example: line_b2?
Response LINE_b2? CLEAR
:
Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the B2 LED is now off and had been off during the current test cycle.
CUR (current) indicates that the B2 indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the B2 LED is off, but had been on during some period before the last query, during the current test cycle.

line_ais? Line Alarm Indicator Signal Query (SONET)

Request the current state of the Alarm Indicator Signal LED.

Example: line_ais?
Response LINE_AIS? CLEAR
:
Note: The section AIS indicator can be in one of three states:
CLEAR indicates that the AIS LED is now off and had been off during the current test cycle.
CUR (current) indicates that the AIS indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the AIS LED is off, but had been on during some period before the last query, during the current test cycle.

line_ferf? Line FERF Query (SONET)

Request the current state of the Far End Receiver Failure alarm LED.

Example: line_ferf?
Response LINE_FERF? CLEAR
:
Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the FERF LED is now off and had been off during the current test cycle.
CUR (current) indicates that the FERF indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the FERF LED is off, but had been on during some period before the last query, during the current test cycle.

line_febe? Line FEBE Query (SONET)

Request the current state of the Far End Block Error alarm LED.

Example: line_febe?

Response: LINE_FEBE? CLEAR

Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the FEBE LED is now off and had been off during the current test cycle.
CUR (current) indicates that the FEBE indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the FEBE LED is off, but had been on during some period before the last query, during the current test cycle.

rs_los? Regeneration Section Loss Signal Query

Request the current state of the LOS indicator.

Example: rs_los?

Response: RS_LOS? CLEAR

Notes: The section LOS indicator can be in one of three states:
CLEAR indicates that the LOS LED is now off and had been off during the current test cycle.
CUR (current) indicates that the LOS indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the LOS LED is off, but had been on during some period before the last query, during the current test cycle.

rs_lof? Regeneration Section Loss Frame Query

Request the current state of the LOF indicator.

Example: rs_lof?

Response: RS_LOF? CLEAR

Note: The section LOF indicator can be in one of three states:
CLEAR indicates that the LOF LED is now off and had been off during the current test cycle.
CUR (current) indicates that the LOF indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the LOF LED is off, but had been on during some period before the last query, during the current test cycle.

rs_oof? Regeneration Section Out-Of-Frame Query

Request the current state of the OOF indicator.

Example: rs_oof?

Response: RS_OOF? CLEAR

Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the OOF LED is now off and had been off during the current test cycle.
CUR (current) indicates that the OOF indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the OOF LED is off, but had been on during some period before the last query, during the current test cycle.

rs_b1? Regeneration Section B1 Query

Request the current state of the section B1 byte indicator.

Example: rs_b1?

Response: RS_B1? CLEAR

Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the B1 LED is now off and had been off during the current test cycle.
CUR (current) indicates that the B1 indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the B1 LED is off, but had been on during some period before the last query, during the current test cycle.

ms_b2? Multiplexer Section B2 Query

Request the current state of the line B2 byte indicator LED.

Example: ms_b2?

Response: MS_b2? CLEAR

Note: The section LOS indicator can be in one of three states:
CLEAR indicates that the B2 LED is now off and had been off during the current test cycle.
CUR (current) indicates that the B2 indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the B2 LED is off, but had been on during some period before the last query, during the current test cycle.

ms_ais? Multiplexer Section Alarm Indicator Signal Query

Request the current state of the Alarm Indicator Signal LED.

Example: ms_ais?
Response: MS_AIS? CLEAR
Note: The section AIS indicator can be in one of three states:
CLEAR indicates that the AIS LED is now off and had been off during the current test cycle.
CUR (current) indicates that the AIS indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the AIS LED is off, but had been on during some period before the last query, during the current test cycle.

ms_rdi? Multiplexer Section Remote Defect Indication Query Cmd

Request the current state of the Multiplexer Section Remote Defect (Degrade) Indication alarm LED.

Example: ms_rdi?
Response: MS_RDI? CLEAR
Note: The section RDI indicator can be in one of three states:
CLEAR indicates that the RDI LED is now off and had been off during the current test cycle.
CUR (current) indicates that the RDI indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the RDI LED is off, but had been on during some period before the last query, during the current test cycle.

ms_rei? Multiplexer Section Remote Error Indicator Query

Request the current state of the Multiplexer Section Remote Error Indicator alarm LED.

Example: ms_rei?
Response: MS_REI? CLEAR
Note: The section REI indicator can be in one of three states:
CLEAR indicates that the REI LED is now off and had been off during the current test cycle.
CUR (current) indicates that the REI indicator is either on or had been on since last queried during the current test cycle.
HIST (history) indicates that the REI LED is off, but had been on during some period before the last query, during the current test cycle.

path_ais? Path AIS Alarm Status

Query - Path AIS alarm condition

Example: path_ais?
Response: Path_ais

path_lop? Path LOP Alarm Status

Query - Path LOP alarm condition

Example: path_lop?
Response: Path_lop

path_rdi? Path RDI Alarm Status

Query - Path RDI alarm condition

Example: path_rdi?
Response: Path_rdi

TOH Transport Overhead Bytes

rx_oh_all? Receive Overhead All Query

Section, Line, Path overhead bytes.

Byte Clear-text

	A1	A2	J0	J1	V1
	B1	E1	F1	B3	V2
	D1	D2	D3	C2	V3
	H1	H2	H3	G1	V4
	B2	K1	K2	F2	
	D4	D5	D6	H4	
	D7	D8	D9	Z3	
	D10	D11	D12	Z4	
	S1	M1	E2	N1	

Example: rx_oh_all?

Response: RX_OH_ALL 0,#HF6;1,#H28;... etc.

Note: The first number before the comma is the NR1 byte count format, ranging from zero to thirty-five for the TOH and POH bytes in the following order: A1, A2, J0, J1, B1, ..., B3, ..., D1, ..., Z5, and thirty-six through thirty-nine for bytes: V1, V2, V3, V4. The #H code after the comma is corresponding byte value in non-decimal numeric (hexadecimal).

rx_oh_byte? [s] Receive Overhead Byte Query

Returns the value of the specified TOH byte.

Whereas the argument [s] is a byte clear-text listed in table above

Example: rx_oh_byte? K2

Response: RX_OH_BYTE K2,#H22

Note: The argument K2 in the example corresponds to the K2 byte. The K2 before the comma in the response in the corresponding byte label. The code #H22 after the comma is the hexadecimal (non-decimal numeric value) of the K2 byte.

rx_oh_offset? [n] Receive Overhead Offset Query

Returns the overhead byte associated with the specified offset. All POH (Path Overhead) and VT (Virtual Tributary) bytes equal zero because only section and line testing are supported by the ST2400A.

Whereas the argument [n] is the corresponding offset value for the byte clear-text using the format *byte clear-text: offset value* listed in table below

Byte Clear-text and Offset

	TOH Section (Sec.) and Line Overhead Bytes			POH Bytes	VT Bytes
Sec.	A1: 00	A2: 01	J0: 02	J1:03	V1: 36
Sec.	B1: 04	E1: 05	F1: 06	B3: 07	V2: 37
Sec.	D1: 08	D2: 09	D3: 10	C2: 11	V3: 38
Line	H1: 12	H2: 13	H3: 14	G1: 15	V4: 39
Line	B2: 16:	K1: 17	K2: 18	F2: 19	
Line	D4: 20	D5: 21	D6: 22	H4: 23	
Line	D7: 24	D8: 25	D9: 26	Z3: 27	
Line	D10: 28	D11: 29	D12: 30	Z4: 31	
Line	S1: 32	M1: 33	E2: 34	Z5: 35	

Example:	rx_oh_offset? 18
Response:	RX_OH_OFFSET #H22
Note:	The number 18 in the example corresponds to the K2 byte's offset value. The number #H22 is the non-decimal numeric (hexadecimal) value of the K2 byte.

J0 Capture and Access Remote Control Implementation

Single byte J0 trace remote access is supported through existing overhead manipulation commands. New 16-byte and 64-byte J0 trace query requirements are supported with the following query / parameters:

Command	**Result**
j0_get? *trace_type*	will get the J0 or J1 multi-byte trace based on the type selected.
j0_get? *16byte*	will get the 16byte ASCII trace and append the “CRC” to the end of the string. *example return value:* ***abcdefghijklmno CRC=AF***
j0_get? *16byte_hex*	will get the 16byte hexadecimal trace (which will be comma separated) and append the “CRC” to the end of the string. *example return value:* ***61,62,63…. CRC=AF***
j0_get? *64byte*	will get the 64byte ASCII trace, no <CR> or <LF> or <NULL> characters will be displayed. *example return value:* ***Tek ST2400A***
j0_get? *64byte_hex*	will get the 64byte hexadecimal trace, (which will be comma separated). *example return value:* ***54,65,6B,20…..,0D,0A***

J0, J1 Trace Capture & Display (RX)

Users are able to set query captured trace messages through either GPIB or RS-232 remote commands.

Users are able to request ASCII trace contents (assumes valid trace containing only printable ASCII chars, nulls, and <CR> <LF>, and skips frame start marker in 16-byte mode), or request 16-byte or 64-byte trace contents in hex. In 16-byte mode the CRC is displayed at the end of the trace in a “CRC=hex value” format.

Command	Result
j1_get? *trace_type*	will get the J0 or J1 multi-byte trace based on the type selected.
j1_get? *16byte*	will get the 16byte ASCII trace and append the "CRC" to the end of the string. *example return value:* ***abcdefghijklmno CRC=AF***
j1_get? *16byte_hex*	will get the 64byte ASCII trace, no <CR> or <LF> or <NULL> characters will be displayed. *example return value:* ***Tek ST2400A***
j1_get? *64byte_hex*	will get the 64byte hexadecimal trace, (which will be comma separated). *example return value:* ***54,65,6B,20…..,0D,0A***

J0 Generation Remote Control Implementation

Single byte J0 trace generation is supported through existing overhead manipulation commands. New 16-byte and 64-byte J0 trace generation requirements are supported with the following command / parameters:

Command:	**Result:**
j0_set *trace_type,comma separated parameters*	will set the J0 multi-byte trace based on the type selected and the parameters passed in.
j0_set *16byte,CRC=FF, abcdefghijklmno*	will set the 16byte trace to the unquoted text string passed in. If more than 15 characters are provided the extra will be truncated. An optional “CRC” can be passed in, and will over-ride the CRC which would normally be calculated automatically.
j0_set *16byte_hex,CRC=FF,00,22,77,..,..*	will set the 16byte trace to the comma separated hexadecimal values passed in. If more than 15 hex values are provided the extra will be truncated. An optional “CRC” can be passed in, and will over-ride the CRC which would normally be calculated automatically.
j0_set *64byte,abcdefghijklmno…*	will set the 64byte trace to the unquoted text string passed in, **the text string must be sent in four different 16 character segments, any extra text will be truncated. The user must specify which segment is being passed on the command-line. *Each segment should be passed in order, i.e first, second, third and fourth*. If more than 14 characters are provided on the “fourth” segment the extra will be truncated.** The <CR> and <LF> characters will be automatically appended to the end, in positions 63 and 64 respectively of the 64byte trace.
j0_set *64byte_hex,00,22,77,..,..*	will set the 64byte trace to the comma separated hexadecimal values passed in, **the values must be sent in four different 16 byte segments, any extra values will be truncated. The user must specify which segment is being passed on the command-line. Each segment should be passed in order, i.e first, second, third and fourth. If more than 16 characters are provided on the “fourth” segment the extra will be truncated.** The <CR> and <LF> characters will be automatically appended to the end (ONLY IF LESS THAN 63 CHARACTERS TOTAL WERE ENTERED), in positions 63 and 64 respectively of the 64byte trace.

Automatic Protection Switching (APS) Data

rx_aps_byte? Receiver APS Bytes Query

Return the current value of the APS (Automatic Protection Switching) bytes: K1 and K2

Example: rx_aps_byte?
Response: RX_APS_BYTE #HFF, #H55
Note: The codes #HFF and #H55 are the hexadecimal numeric value of the K1 and K2 bytes, respectively.

Receiver Drop Commands

rx_drop [off | 155 | 622] Receiver Drop Rate Set

Sets the optical drop rate.

Arguments: off: set the optical drop to off and enables the 155 Mb/s drop.
155: sets the optical drop to 155 Mb/s and enables the 155 Mb/s drop.
622: sets the optical drop to 622 Mb/s and disables all drop.
Example: rx_drop 155
Response: NONE
Note: The receiver setup is independent of the transmitter setup.

rx_drop? Receiver Drop Rate Query

Returns the value of the current optical drop rate selected.

Example: rx_drop?
Response RX_DROP 155
:
Note: See the Receiver Drop Rate Set Command.

drop_chan n Drop Channel Tributary Select

Selects the tributary channel to drop.

Arguments: n = from 1 to 4 if the drop rate is set to 622 Mb/s
n = from 1 to 16 if the drop rate is set to 155 Mb/s
Example: drop_chan 5
Response: NONE
Note: The receiver setup is independent of the transmitter setup.

drop_chan? Drop Channel Tributary Query

Returns the channel number of the tributary drop selected.

Example: drop_chan?
Response: DROP_CHAN 5
Note: See the Drop Channel Tributary Select Command

lof_dur? LOF duration - APS counts

Returns the Loss of Frame duration, in milliseconds

Example: lof_dur?
Response: LOF_DUR 1200 ms

This condition is cleared everytime a new LOS event occurs or the RUN/STOP key is pressed to start a new test.

los_dur? LOS duration - APS counts

Returns the the Loss of Signal duration, in milliseconds

Example: los_dur?
Response: LOF_DUR 1200 ms

This condition is cleared everytime a new LOS event occurs or the RUN/STOP key is pressed to start a new test.

Transmitter Remote Commands

If a GPIB command is entered, but the appropriate hardware for that command is not present in the ST2400A test set, the command will not execute. Certain commands are invalid in Transmitter-only or Receiver-only configurations.

tx_2488_elec [on | off]

Set the "ON/OFF" state of the Transmitter electrical output

Example:	tx_2488_elec on

tx_2488_elec?

Query the "ON/OFF" state of the Transmitter electrical output

Example:	tx_2488_elec?
Response	ON

tx_event_trig [alarm | error]   Event Trigger Set

Selects the trigger stimulus for the rear-panel event trigger output.

Arguments:	alarm: generates a trigger on each occurrence of the currently configured alarm. error: generates a trigger on each occurrence of the currently configured error.
Example:	tx_event_trig alarm
Note:	The event trigger will lead the preset alarm by less than 1 microsecond. It is available at the rear panel and can be used to synchronize other test equipment.

tx_event_trig?   Event Trigger Query

Returns the value of the selected event trigger stimulus, alarm or error.

Example:	tx_event_trig alarm
Note:	See the Event Trigger Set Command.

tx_ext_sync [off | 1544 | 2048]

Sets 1.544 Mbit/s, 2Mbit/s/2 MHz inputs for external timing synchronization

Argument	tx_ext_sync off	Disables external sync input
	tx_ext_sync 1544	Enable 1.544 Mbit/s external sync input
	tx_ext_sync 2048	Enable 2.048 Mbit/s/2 MHz external sync input
Example:	tx_ext_sync 2048	

tx_ext_sync?

Queries the current setting. Used to add 1.544 Mbit/s, 2Mbit/s/2 MHz inputs for external timing synchronization

Example:	tx_ext_sync?

tx_input [155 | 622 | 155e | int | thru] Tx external input selection command

Sets the source, rate, and configuration that will produce the final transmitter output

Arguments:	155: configures the transmitter output from the transmitter 155 Mb/s external optical input 622: configures the transmitter output from the transmitter external 622 Mb/s optical input 155e: Set external trib channel to 155 electrical (to match front panel label) int: configures the transmitter output from an internal generator thru: configures the transmitter output from the receiver input
Example:	tx_input 155
Response	NONE
Note:	The transmitter setup is independent of the receiver setup.

tx_input? TX input query command

Returns the status of the transmitter source, rate, and configuration.

Example:	tx_input?
Response:	TX_INPUT xxx
Note:	The response can be 155, 622, int, or thru (see tx_input command)

tx_laser laser,[off | on | cw]

Set the laser to "ON" with modulation, "CW" without modulation or "OFF" state of operation for the specified LASER module

Argument	tx_laser 1,on	Set laser module ON with modulation
	tx_laser 1,off	Set laser module OFF
	tx_laser 1,cw	Set laser module ON without modulation
Example:	tx_laser 1,on	
Note	**tx_laser** is an equivalent command. **tx_laser** is recommended for the ST2400A test unit. Only use **tx_laser_pow** for backward compatibility with older ST2400 test units. The CW choice is not available for the tx_laser_pow command.	

tx_laser?

Queries operation "OFF/ON/CW" setting for the specified LASER Module

Example: tx_laser? 1
Response CW
Note: ON indicates that the transmitter laser is enabled with modulation.
OFF indicates that the transmitter laser is disabled.
CW indicates that the transmitter laser is enabled without modulation
The Transmitter setup is independent of the Receiver setup.
Note: **tx_laser** is an equivalent command for **tx_laser** laser. **tx_laser** is recommended for the ST2400A test unit. Only use **tx_laser_pow** for backward compatibility with older ST2400 test units. The CW choice is not available for the tx_laser_pow command.

tx_laser_info? [laser]

Queries the selected "laser" (if the selected laser is installed) and displays a detailed description in the following format: **board_type, rev_level, options, cal_date.**

Example: tx_laser_info? 1
Return Value 0x10, A1, 1, 06/01/98

tx_laser_pow [on | off]

Sets the laser selected by the **tx_wavelen** command to "ON" or "OFF".

Example: tx_laser_pow on
Response: NONE
Note The Transmitter setup is independent of the Receiver setup.
Note **tx_laser** and **tx_laser_pow** are equivalent commands.
tx_laser is recommended for the ST2400A test

tx_laser_pow?

Queries operation of laser selected by the **tx_wavelen** command.

Example: tx_laser_pow?
Response: TX_LASER_POW ON
Note: ON indicates that the transmitter laser is enabled.
OFF indicates that the transmitter laser is disabled.
The Transmitter setup is independent of the Receiver setup.
Note: **tx_laser_pow?** and **tx_laser?** are equivalent commands.
tx_laser? is recommended for the ST2400A test set

tx_laser_type? [laser]

Queries the selected "laser" (if the selected laser is installed) and displays the selected laser's type in a plain text format based on SONET or SDH mode, whichever is selected. The type formats are listed in the following table.

Example: tx_laser_type? 1
Return Value L-16.2

The types displayed for each of the laser modules are as follows:

Laser Module	SONET Std	SONET Type	SDH Std	SDH Type
1310nm DM	GR-253	LR-1	G.957	L-16.1
1550nm DM	GR-253	LR-2	G.957	L-16.2

tx_laser_wl? [laser]

Queries the selected "laser" (if the selected laser is installed) and displays the selected laser's wavelength in either (precise) or (non-precise) format based on customer-ordered options

Example: tx_laser_wl? 1
Return Value 1550.12 (precise)
1550 (non-precise)
Note: tx_wavelen? and laser_wl? are equivalent commands.
laser_wl? is recommended for the ST2400A test set

tx_mapping [sts1 | sts3c | vc4 | sts12c | sts48c | vc4-4c | vc4-16c]

Sets structured payload mapping.

Argument tx_mapping? Query current setting - Any mode
tx_mapping sts1 Set mapping to STS1 (SONET only, any ST2400, with new TX)
tx_mapping sts3c Set mapping to STS3C (SONET only, any ST2400, with new TX)
tx_mapping sts12c Set mapping to STS12C (SONET only, with OC12/48c hardware)
tx_mapping sts48c Set mapping to STS48C (SONET only with OC12/48c hardware)
tx_mapping vc4 Set mapping to VC4 (SDH only, any ST2400 with new TX)
tx_mapping vc4-4c Set mapping to VC4-4C (SDH only with OC12/48c hardware)
tx_mapping vc4-16c Set mapping to VC4-16c (SDH only with OC12/48c hardware)

Example: tx_mapping vc4-4c

tx_mapping?

Queries the current setting, structured payload mapping.

Example: tx_mapping?
Response TX_MAPPING VC4

tx_sync_los?

Queries the "tx_ext_sync" Loss of Signal condition

Example:	tx_sync_los? 1
Return values	OK, 1544 LOS, 2048 LOS

tx_test_patt [pn23 | pn23inv | allzeros | fixed= 7F | pn31 | pn31inv]

Queries the current setting, sets payload pattern choice.

Argument	tx_test_patt pn23	Set mode to PN23
	tx_test_patt pn23inv	Set mode to PN23 Inverted (O.181)
	tx_test_patt allzeros	Set mode to ALL-ZEROS
	tx_test_patt fixed= 7F	Will set pattern to entered hexidecimal value

Note: tx_test_patt fixed= 00 is equivalent to tx_test_patt allzeros

	tx_test_patt pn31	Set mode to PN31
	tx_test_patt pn31inv	Set mode to PN31 Inverted

Example: tx_test_patt pn23

tx_test_patt?

Queries the current setting, payload pattern choice.

Example:	tx_test_patt?
Response	TX_TEST_PATT PN23

tx_wavelen [1310 | 1550]

Sets the laser wavelength on older ST2400 units or the closest laser matching the chosen wavelength on newer ST2400A units.

Example:	tx_wavelen 1550
Argument	tx_wavelen 1550 tx_wavelen 1310
Note	If the wavelength chosen does not match the laser, the setting will default to a wavelength that does exist. **tx_laser_wl** is recommended for the ST2400A test set

tx_wavelen?

Queries the laser wavelength on older ST2400 units or the closest laser matching the chosen wavelength on newer ST2400A units.

Example:	tx_wavelen?
Response	Valid response values: 1310, 1550
Note:	**tx_wavelen?** and **tx_laser_wl?** are equivalent commands. **Tx_laser_wl?** is recommended for the ST2400A test set

tx_wl_offset [laser, -50 to 49]

Sets the wavelength offset for the specified "laser" module. Range is -50 to 49.

Example:	tx_wl_offset 1,49

tx_wl_offset?

Queries wavelength offset setting for the specified "laser" module

Example:	tx_wl_offset 1
Return Value	-50

External tributary insertion channel commands

tx_trib_mode [single | all]   TX Mode Select

Selects the multiplex mode that will be used for the 2.4 Gb/s signal.

Arguments:	single: will insert the 155 Mb/s tributary selected into the into the channel specified by the tx_trib_chan command. all: will insert the 155 Mb/s tributary selected into all 16 (or the 622 Mb/s tributary selected into all 4) channels in the 2.4 Gb/s transmitter output signal.
Example:	tx_trib_mode single
Response:	NONE
Notes:	This command is used with the tx_trib_chan command. The receiver setup is independent of the transmitter setup.

tx_trib_chan n   TX Channel Set

Selects the channel that will be used as the tributary to be multiplexed with the 2.4 Gb/s signal.

Arguments	n = from 1 to 4 if the drop rate is set to 622 Mb/s. n = from 1 to 16 if the drop rate is set to 155 Mb/s.
Example:	tx_trib_chan n
Response:	NONE
Notes:	This command is external drop rate dependent, as set by the tx_input command. When executed, this command will generate an ERROR response if the input is not set to one of the external modes (see the tx_input command). The receiver setup is independent of the transmitter setup.

Single error insertion commands

tx_allchan b2

Inject a single B2 error rate over payload channels simultaneously

Example:	tx_allchan b2
Note:	Not available for B3 boards/Option 42

tx_serr [b1 | b2 | b3 | pyld_bit| chan | b2_rem | rei | tse]

Injects a single error of the specified type.

Arguments: b1: inverts a single bit in the b1 byte to generate a parity error
b2: inverts a single bit in the b2 byte to generate a parity error
b2: generates a single B2 error on a single channel
b3: generates a single B3 error on a single channel
pyld_bit: generate a pattern bit error on a single channel *
chan: inserts a bit error on the channel specified by the tx_err_chan command.
b2_rem works only for older ST2400 units only:
b2_rem: inserts a single b2 error when the transmitter is in the external mode with tributary insertion on channel one or on all of the channels.
This options only functions in the dual mode.
rei: inserts a remote error indication (rei)

Example: tx_serr b2

Response: NONE

Note: Used with the tx_err_chan command
*** tx_serr pyld_bit** can also be shown as **tx_serr tse** (for SDH)
bit and chan not available for B3 boards/ Option 42

tx_err_chan n

Sets the channel for a bit error on a specific STM-1 channel.

Example: tx_err_chan n (n = an integer from 1 to 16)

Response: NONE

Note: Used with the tx_serr command
The channel will be used for a single or rate error insertion.

Error Rate Commands

tx_err_rate x.yE-z   Error Rate Generator Range Set

Sets the rate generator range for bit, channel, and B1, B2, B3, payload bit errors.

Example: tx_err_rate x.yE-z
Response: NONE
Note: Bit and channel errors can be set to a range
from 0.1e-9 to 1.0e-3
B1 errors can be set to a range from 0.1e-9 to 1.0e-5
B2 errors can be set to a range from 0.1e-9 to 1.0e-3
B3 errors can be set to a range from 0.1e-9 to 1.0e-3
See the Error Rate Injection Enable Command

tx_err_rate? Error Rate Generator Range Set

Returns the current error rate settings.

Example: tx_err_rate?
Response: TX_ERR_RATE x.yE-z
Note: See the Error Rate Generator Range Set Command

tx_err_ena [b1 | b2 | b3 | pyld_bit | tse | bit | chan | off]   Error Rate Injection Enable

Enables or disables error rate insertion.

Arguments: b1: generates b1 errors at the current rate setting
b2: Inject a B2 error rate on all payload channels in rotation
b3: Inject the error rate into the selected channel
pyld_bit: Inject the error rate into the selected channel
chan: generates channel errors at the current rate setting
off: disables the error that are being generated at the current rate.
Example: tx_err_ena b2
Response: NONE
Notes: The current rate was set by the Error Rate Generator Range Set Command

tx_err_ena?   Error Rate Injection Enable Query

Returns the status of the Error Rate Injection Enable Command

Example: tx_err_ena?
Response
: TX_ERR_ENA B1
Notes: The current status can be b1, b2, b3, tse, pyld-bit, chan, or off.

tx_febe_cnt [n]   Tx Line FEBE Errored Frames Count Set (SONET)

Sets the number [n] of Line FEBEs to be inserted per frame.

Arguments	n is a number from 1 to 255
Example:	tx_febe_cnt 14
Response:	NONE
Note:	Used with the burst argument of the tx_febe_ena command.

tx_febe_cnt?   Tx FEBE Errored Frames Count Query (SONET)

Returns the number of Line FEBE Signals set to be inserted in each frame.

Example:	tx_febe_cnt?
Response:	TX_FEBE_CNT 14 See the Tx Line FEBE Errored Frames Count Set Command.

tx_febe_secs [n.m]   Tx FEBE Duration Set (SONET)

Sets the Line FEBE duration of time in seconds.

Argument:	n.m is a decimal number from 0.1 to 9.9 seconds.
Example:	tx_ febe_secs 3.9
Response:	NONE
Notes:	The default value is 2.5 seconds. Used with the timed argument of the tx_febe_ena command.

tx_febe_secs?   Tx FEBE Duration Query (SONET)

Returns the current settings for the Line FEBE duration in seconds.

Example:	tx_febe_secs?
Response:	TX_FEBE_SECS 3.9
Note:	See the Tx FEBE Duration Set Command

tx_febe_ena [burst | timed | cont | off] (SONET)

Enables or disables the far end block error (FEBE) insertion event.

Arguments:	burst generates one single error event timed generates the event for the period set cont generates a continuous event off disables the event including continuous and burst
Example:	tx_febe_ena off
Response	NONE
Note:	The duration of the timed event is set by the tx_febe_secs command

tx_febe_ena? Tx FEBE Enable Query (SONET)

Returns the current status FEBE insertion event.

Example:	tx_febe_ena? off
Response:	TX_FEBE_ENA OFF
Note:	See the Tx FEBE Enable Set Command

tx_rei_cnt [n] Tx REI Errored Frames Count Set

Sets the number [n] of frames that will contain a Line REI (Remote Error Indication) signal.

Argument:	n is a number from 1 to 255
Example:	tx_rei_cnt 14
Response:	NONE
	Used with the burst argument of the tx_rei_ena command.

tx_rei_cnt? Tx REI Errored Frames Count Query

Returns the number of Line REI Signals set to be inserted in each frame.

Example:	tx_rei_cnt?
Response:	TX_REI_CNT 14
	See the Tx REI Errored Frames Count Set Command.

tx_rei_secs [n.m] Tx REI Duration Set

Sets the duration of the Line REI (Remote Error Indication) insertion time in seconds.

Arguments:	n.m is a decimal number from 0.1 to 9.9 seconds.
Example:	tx_rei_secs 3.9
Response:	NONE
Notes:	The default value is 2.5 seconds.
	Used with the timed argument of the tx_rei_ena command.

tx_rei_secs? Tx REI Duration Query

Returns the current settings for the Line REI duration in seconds.

Example:	tx_rei_secs?
Response:	TX_REI_SECS 3.9
Note:	See the Tx REI Duration Set Command

tx_rei_ena [burst | timed | cont | off] Tx REI Enable Set

Enables or disables the REI (Remote Error Indication) insertion event.

Arguments: burst: generates one single error event for a number of frames.
timed: generates the event for the period set
cont: generates a continuous event
off: disables the event including continuous and burst

Example: tx_rei_ena off

Response: NONE

Note: The number of errored frames is set by the tx_rei_cnt command
The duration of the timed event is set by the tx_rei_secs command

tx_rei_ena? Tx REI Enable Query

Returns the current status REI insertion event.

Example: tx_rei_ena off

Response: TX_REI_ENA OFF

Note: See the Tx REI Enable Set Command

TOH Edit and Query Commands

The Transport Overhead edit commands tx_oh_byte, tx_oh_offset, tx_oh_pass, and tx_oh_write can only be used after the tx_oh_edit on command. Either the tx_oh_edit enter or tx_oh_edit esc command should be used to end the editing session with or without saving. Query commands can be used independently.

M0 is the first Z2 byte, and is defined only for STS-1s (SONET). The
ST2400 allows the first Z2 byte to be displayed. The displayed Z2 byte is the M0 byte for an STS-1 in Channel 1.

tx_oh_edit [on | esc | enter]   TX Overhead Edit

This command is used to start and end, with or without saving the changes, the TOH editing session.

Arguments:	on: starts the editing session esc: ends the editing session without saving any changes enter: saves all changes and ends the edition session
Example:	tx_oh_edit on
Response:	NONE
Note:	This command enables and disables the use of the following commands: tx_oh_byte, tx_oh_offset, tx_oh_pass, and tx_oh_write

tx_oh_edit?   TX Overhead Query

Returns status of the TOH editing session.

Example:	tx_oh_edit?
Response:	TX_OH_EDIT ON
Note:	The editing session can be either ON or OFF.

tx_oh_all?   Transmit Overhead All Query

Returns 36 overhead bytes that include all section, line, and path overhead bytes.

Example:	tx_oh_all?
Response:	TX_OH_ALL [n],[b]; [n],[b]; [n],[b]; ... [n],[b]; [n],[b]
Note:	n = the byte number from 0 to 39 and b = the byte content in hexadecimal format, such as 0,#HF6; 1,#H28; ... etc.

tx_oh_def   Transmit Overhead Default

Sets the TOH (transmit overhead)bytes to factory default settings. This command only affects the bytes which the user may edit.

Example:	tx_oh_def
Response:	NONE
Note:	The factory default settings are 0 for all TOH (transmit overhead) bytes except E1 and E2 which are set to 7F in hexadecimal format. This command has no effect in an edit session (see th_oh_edit).

tx_oh_byte? ["s"]   Transmit Overhead Byte Query

Returns the current value of the overhead byte [s].

Example: tx_oh_byte? ["s"]
Response: TX_OH_BYTE [s], [b]
Note: s = the byte clear-text from A1 to Z5 and b = the byte content in hexadecimal format, such as K2,#H22

Clear-text

	TOH Section (Sec.) and Line Overhead Bytes			POH Bytes	VT Bytes
Sec.	A1	A2	J0	J1	V1
Sec.	B1	E1	F1	B3	V2
Sec.	D1	D2	D3	C2	V3
Line	H1	H2	H3	G1	V4
Line	B2	K1	K2	F2	
Line	D4	D5	D6	H4	
Line	D7	D8	D9	Z3	
Line	D10	D11	D12	Z4	
Line	S1	M1	E2	Z5	

tx_oh_byte ["s"], [b]   Transmit Overhead Byte Set

Sets the specified TOH byte to any hexadecimal number between 00 and FF Hex

Example: tx_oh_byte [s],[b]
Response: NONE
Note: s = the byte clear-text from A1 to Z5 and b = the byte content in hexadecimal format, such as K2,#H22
Enabled and disabled by the tx_oh_edit command

Byte Clear-text

	TOH Section (Sec.) and Line Overhead Bytes			POH Bytes	VT Bytes
Sec.	A1	A2	J0	J1	V1
Sec.	B1	E1	F1	B3	V2
Sec.	D1	D2	D3	C2	V3
Line	H1	H2	H3	G1	V4
Line	B2	K1	K2	F2	
Line	D4	D5	D6	H4	
Line	D7	D8	D9	Z3	
Line	D10	D11	D12	Z4	
Line	S1	M1	E2	Z5	

tx_oh_offset [n],[b] Transmit Overhead Offset Byte Set

Sets the overhead byte associated with offset (n) to any hexadecimal value (b) from 00 to FF

Example: tx_oh_offset 1,AA
Response: NONE
Notes: [n],[b] (n = offset and b = the byte in hexadecimal format)
The table below list each TOH byte clear-text and its offset.
Enabled and disabled by the tx_oh_edit command

Byte Clear-text and Offset

	A1: 00	A2: 01	J0: 02	J1:03	V1: 36
	B1: 04	E1: 05	F1: 06	B3: 07	V2: 37
	D1: 08	D2: 09	D3: 10	C2: 11	V3: 38
	H1: 12	H2: 13	H3: 14	G1: 15	V4: 39
	B2: 16:	K1: 17	K2: 18	F2: 19	
	D4: 20	D5: 21	D6: 22	H4: 23	
	D7: 24	D8: 25	D9: 26	Z3: 27	
	D10: 28	D11: 29	D12: 30	Z4: 31	
	S1: 32	M1: 33	E2: 34	Z5: 35	

tx_oh_offset? [n] Transmit Overhead Offset Query

Returns the overhead byte associated with offset [n]

Example: tx_oh_offset? [n]
Response: TX_OH_OFFSET [n],[b] (n = offset and b = the byte in hexadecimal format)
Note: offset mapping is shown with the tx_oh_offset [n],[b] command.

tx_oh_pass [1...9 | all] Transmitter Overhead Pass

Sets the overwrite/pass attribute of the specified group to pass

Argument s: 1 = the E1 byte; 2 = the F1 byte; 3 = bytes D1 through D3; 4 = bytes K1 and K2; 5 = bytes D4 through D12; 6 = byte S1; 7 = byte M1; 8 = byte E2; 9= byte J0; or all = bytes listed.
Example: tx_oh_pass 4
Response: NONE
Note: Enabled and disabled by the tx_oh_edit command

tx_oh_write [1...9 | all] Transmitter Overhead Write

Sets the attribute of the specified group to overwrite.

Argument s: 1 = the E1 byte; 2 = the F1 byte; 3 = bytes D1 through D3; 4 = bytes K1 and K2; 4 = bytes D4 through D12; 6 = byte S1; 7 = byte M1; 8 = byte E2; 9= byte J0; or all = bytes listed.

Example: tx_oh_write 4

Response: NONE

Note: Enabled and disabled by the tx_oh_edit command

tx_oh_b_pass [j0 | e1 | f1 | d1 | k1 | d4 | m1 | s1 | e2 | c2 | all]

Sets the overwrite/pass attribute of the specified group to pass.

Example: tx_oh_b_pass k1

Response : NONE

Note: Uses character names for the bytes instead of numbering the groups.
d1 sets d1...d3, k1 sets k1 & k2, d4 sets d4...d12
Returns “NONE” when queried if no bytes are in overwrite mode.
Enabled and disabled by the tx_oh_edit command

tx_oh_b_write [j0 | e1 | f1 | d1 | k1 | d4 | m1 | s1 | c2 | e2 | all]

Sets the attribute of the specified group to overwrite.

Example: tx_oh_b_write d1

Response : NONE

Note: Uses character names for the bytes instead of numbering the groups.
d1 sets d1...d3, k1 sets k1 & k2, d4 sets d4...d12
Returns “NONE” when queried if no bytes are in overwrite mode.
Enabled and disabled by the tx_oh_edit command

Repeat Errors

re_type [b1 | b2 | b3 | pyld_bit | l_febe *or* rei | chan], Set Repeat Error Type

Sets the recurring error type to be injected.

Arguments	b1: b1 error b2: b2 error b3: b3 error pyld_bit: Payload bit error l_febe or rei chan: channel error
Example:	re_type b1
Response:	NONE
Note:	Only one error can be active. Enabling any error will disable all others. Errors are injected a rate for an period of time. Repeat errors use the rate set by each individual transmit error command listed under STM-16 error rate commands.
Note:	bit and chan not available for B3 boards/ Option 42

re_type? Repeat error Query

Query error for the error type selected.

Example:	re_type?
Response:	RE_TYPE? [error type]
Note:	Error types are listed with the Set Repeat Error Type Command

re_inject [n.m] Repeat Error Inject

Sets the on-time for the recurring error injection period from 0.1 to 9.9 seconds

Example:	re_inject [n.m]
Response :	NONE

re_inject? Repeat Error Inject Query

Returns the on-time for the recurring error injection period (cycle).

Example:	re_inject?
Response :	RE_INJECT [n.m] (n.m = seconds in decimal from 0.1 to 9.9)

re_wait [n.m] Repeat Error Wait

Sets the off-time for the recurring error injection period from 1.0 to 30.0 seconds

Example:	re_inject [n.m]
Response :	NONE

re_wait? Repeat Error Wait Query

Returns the off-time for the recurring error injection period (cycle).

Example:	re_wait?
Response :	RE_WAIT [n.m] (n.m = seconds in decimal from 1.0 to 30.0)

re_enable [on | off] Repeat Error Enable Set

Controls the repeat error cycle using preset conditions.

Arguments:	on: activates the repeat error cycle off: halts the current repeat error cycle
Example:	re_enable on
Response:	NONE
Note:	Before using this command, the preset conditions can be review using the Repeat Error Query commands: re_type?, re_inject? and re_wait?

re_enable? Repeat Error Enable Query

Returns the status of the Repeat Error Enable Set Command

Example:	re_enable?
Response:	RE_ENABLE ON
Note:	The Repeat Error Enable Set Command can be either ON or OFF

Alarms

alm_clear Alarm Clear

Turns off all alarm conditions in progress and disables all continuous alarm conditions.

Example: alm_clear
Response: NONE
Note: This command has the same function as the alm_enable off command.

alm_enable [burst | multi | timed | off] Alarm Enable

Executes or stops the execution of the alarm conditions that were configured by other commands.

Arguments: burst: generates one single alarm event set by the alm_count command
multi: generates continuous alarm events
timed: generates a timed alarm set by the alm_secs command.
off: stops all alarms events including continuous and burst
Example: alm_enable burst
Response: NONE
Note: Alarm events include all continuous alarm conditions and current burst alarm conditions if the burst is in progress.

alm_enable? Alarm Enable Query

Returns the status of the alarm condition set by the Alarm Enable Set Command

Example: alm_enable?
Response: ALM ENABLE MULTI
Note: See the Alm Enable Set Command.

alm_burst [oof | rdi | l_ais | clr]   Alarm Burst

Specifies which alarm condition will be generated when the alarm enable is set to burst.

Argument:	oof: Out Of Frame rdi: inserts a Remote Defect Indication l_ais: Line Alarm Indication Signal clr: clears the burst alarm condition
Example:	alm_burst oof
Response:	NONE
Note:	Only one alarm condition at a time can be generated in the burst mode. See the alm_enable command.

alm_burst?   Alarm Burst Query

Returns the status of the alarms set by the Alarm Burst Set Command

Example:	alm_burst?
Response:	ALM_BURST OOF
Note:	Only one alarm condition at a time can be generated in the burst mode. See the alm_enable command.

alm_timed [oof | rdi | l_ais | clr]   Alarm Timed Set

Specifies which alarm condition will be generated when the alarm mode is set to timed.

Argument:	oof: Out Of Frame rdi: inserts a remote defect indication (rdi) l_ais: Line Alarm Indication Signal clr: clears the burst alarm condition
Example:	alm_timed rdi
Response:	NONE

alm_timed?   Alarm Timed Query

Returns the status that was set by the Alarm Timed Set Command

Example:	alm_timed?
Response:	ALM_TIMED CLR
Note:	See the Alarm Timed Set Command.

alarm_multi [oof | rdi | l_ais | clr]   Alarm Multiple Set

Specifies which alarm condition will be generated when the alarm mode is set to multiple.

Argument: oof: Out Of Frame
rdi: inserts a remote defect indication (rdi)
l_ais: Line Alarm Indication Signal
clr: clears the burst alarm condition
Example: alarm_multi l_ais
Response: NONE

alm_multi?   Alarm Multiple Query

Returns the status that was set by the Alarm Multiple Set Command
Example: alm_multi?
Response: ALM_MULTI CLR
Note: See the Alarm Multiple Set Command.

alm_count [n]   Alarm Count

Sets the alarm burst duration to on for a specified number of frames from 1 to 127
Example: alm_count 40
Response: NONE
Notes: Use the alm_enable burst command to executed the alarm condition.
The default is 5 frames.

alm_count?   Alarm Count Query

Returns the current valu of the alarm burst duration for the specified number of frames.
Example: alm_count?
Response
:
ALM_COUNT 40
Note: See the Alarm count Set Command

alm_secs [n.m]   Alarm Seconds Set

Sets timed alarm duration in seconds
Argument: n.m: is from 0.1 to 9.9 seconds. seconds.
Example: alm_secs 2.5
Response: NONE
Note: n.m defaults to 2.5 seconds during the power–on sequence.

alm_secs? Alarm Seconds Query

Returns the set duration of the Timed Alarm in seconds

Example: alm_secs?
Response: ALM_SECS 2.5
Note: See the Alarm Seconds Set Command

Repeat Alarms

ra_type [los | oof | rdi | l_ais] Repeat Alarm Select

Specifies which alarm condition will be generated, repetitively, when started.

Argument:	los: inserts a Loss Of Signal oof: inserts an Out Of Frame Signal rdi: inserts a Remote Defect Indication l_ais: inserts a Line Alarm Indication
Example:	ra_type los
Response:	NONE
Note	Used with the ra_start command.

ra_type? Repeat Alarm Select Query

Returns the specific alarm condition to be generated, repetitively, when started.

Example:	ra_type?
Response:	RA_TYPE LOS
Note:	Used with ra_type command

ra_inject [n.m] Repeat Alarm Inject Set

Sets the amount off alarm injection time from 0.1 to 1.0 seconds

Argument:	n.m = a decimal number from 0.1 to 1.0
Example:	ra_inject 0.4
Response:	NONE

ra_inject? Repeat Alarm Inject Query

Returns the currently set Repeat Alarm injection time (on-time) in seconds

Example:	ra_inject?
Response:	RA_INJECT 0.4
Note:	See the Repeat Alarm Inject Set Command

ra_wait [n.m] Repeat Alarm Wait Set

Sets the amount off wait time (off-time) between alarm injections from 9.9 to 30.0 seconds

Argument:	n.m = a decimal number from 9.9 to 30.0
Example:	ra_wait 15.5
Response:	NONE

ra_wait? Repeat Alarm Wait Query

Returns the currently set Repeat Alarm wait time (off-time) in seconds

Example: ra_wait?
Response
: RA_WAIT 15.5
Note: See the Repeat Alarm Wait Set Command

ra_enable [on | off] Repeat Alarm Enable Set

Controls the repeat alarm cycle using preset conditions.

Arguments: on: activates the repeat alarm cycle
off: halts the current repeat alarm cycle
Example: ra_enable on
Response: NONE
Note: Before using this command, the preset conditions can be review using the Repeat Alarm Query commands: ra_type?, ra_inject? and ra_wait?

ra_enable? Repeat Alarm Enable Query

Returns the status of the Repeat Alarm Enable Set Command

Example: ra_enable?
Response: RA_ENABLE ON
Note: The Repeat Alarm Enable Set Command can be either ON or OFF

Loss Of Signal

los_enab Loss Of Signal Enable

Enables the LOS event for the amount of time specified by los_time.

Example:	los_enab
Response:	NONE

los_time n Loss Of Signal Time

Set Loss of Signal event to a specific amount of time from 1 to 127 microseconds

Example:	los_time 50
Response:	NONE
Note:	In the example the LOS will last for 50 microseconds

los_time? Loss Of Signal Time Query

Returns the LOS time duration in microseconds

Example:	los_time?
Response:	LOS_TIME n (n = a decimal quantity from 1 to 127 microseconds)

los_secs [n.m] Loss Of Signal Set

Sets the timed LOS event to a duration from 1.0 to 9.9 seconds

Argument	n.m = a decimal number from 1.0 to 9.9
Example:	los_secs 4.0
Response:	NONE

los_secs? Loss Of Signal Query

Returns the duration in seconds that is set for the timed LOS event

Example:	los_secs?
Response:	LOS_SECS 4.0
Note:	See the Loss Of Signal Set Command

los_tm_enab Loss Of Signal Enable

Enables the LOS event for the amount of time specified by los_secs command.

Example:	los_tm_enab
Response:	NONE

los_off Loss Of Signal Disable

Disables the LOS event currently being generated.

Example:	los_off
Response:	NONE

los_cont

Generates a continuous LOS event.

Example:	los_cont
Response:	NONE

los_active?

Queries the current condition of the LOS event.

Example:	los_active?
Response:	LOS_ACTIVE ON
Note:	ON indicates that the LOS event is currently activated, OFF indicates that the LOS event is inactive.

Combined (Dual) Mode Commands

When the ST2400A is connected to a Tektronix ST112, the combined (dual) mode on command can be used to initiate the link procedure between the two systems. When the link is made the ST2400A will be controlled by the ST112 through its remote serial interface (RS232C port). All remote commands sent to the ST2400A through the RS-232C port will be passed on to the ST112 for processing.

When the link is already established, the combined mode off command will be handled by the ST112, which will terminate the connection between the two systems.

NOTE: When remote commands are sent through the ST2400A GPIB ports, the ST2400A will process them, even if the ST2400A is in the combined (dual) mode.

combined_mode [on | off]   Combined Mode Set

Used to initiate or terminate the combined mode between the ST2400A and ST112

Example: combined_mode on
Response NONE
:

combined_mode?   Combined Mode Query

Used to query the combined mode status, between the ST2400A and ST112

Example: combined_mode?
Response COMBINED_MODE OFF
:

Appendices

LabView for ST2400A

LabView is an instrument driver providing programming support for the ST2400A test set. This instrument driver contains functions for opening, configuring, taking measurements from, and closing the instrument in field and lab environments.

Requirements to use LabView

To successfully use this module, the following conditions must be met.

For GPIB instrument drivers

- The instrument is connected to the GPIB interface
- The GPIB address supplied to initialize function must match the GPIB address of the instrument.

For VXI instrument drivers

- The instrument is installed in the VXI mainframe and one of the following controller options is being used - Embedded controller; MXI; MXI2; GPIB-VXI
- The logical address supplied to the initialize function must match the logical address of the instrument.

For RS-232 instrument drivers.

- The instrument is connected to the RS-232 interface.
- The COM port, baud rate, parity and timeout supplied to the initialize function must match the settings of the instrument.

Diskette Contents

See the *Readme.txt* file on the diskette for a listing of files and installation instructions.

Installation Instructions

See the *Readme.txt* file on the diskette for a listing of files and installation instructions.

Viewing Documentation

Complete documentation on how to use these instrument drivers is available from the software itself.

In LabView, go to *File: Print Documentation*.

In LabWindows, *Open the Function Tree (Ctrl + O)*. From the Function Tree, choose *Options: Generate Documentation*. This action will create a document labeled **tks2400.doc.** This document describes each function in the instrument. The functions are listed in alphabetical order with a description of the function, C syntax of the function, a description of each parameter, and possible error codes.

LabWindows for ST2400A

LabWindows is an instrument driver providing programming support for the ST2400A test set. This instrument driver contains functions for opening, configuring, taking measurements from, and closing the instrument in field and lab environments.

Requirements to use LabWindows

To successfully use this module, the following conditions must be met.

For GPIB instrument drivers

- The instrument is connected to the GPIB interface
- The GPIB address supplied to initialize function must match the GPIB address of the instrument.

For VXI instrument drivers

- The instrument is installed in the VXI mainframe and one of the following controller options is being used - Embedded controller; MXI; MXI2; GPIB-VXI
- The logical address supplied to the initialize function must match the logical address of the instrument.

For RS-232 instrument drivers.

- The instrument is connected to the RS-232 interface.
- The COM port, baud rate, parity and timeout supplied to the initialize function must match the settings of the instrument.

Diskette Contents

See the *Readme.txt* file on the diskette for a listing of files and installation instructions.

Installation Instructions

See the *Readme.txt* file on the diskette for a listing of files and installation instructions.

Viewing Documentation

Complete documentation on how to use these instrument drivers is available from the software itself.

In LabView, go to *File: Print Documentation*.

In LabWindows, *Open the Function Tree (Ctrl + O)*. From the Function Tree, choose *Options: Generate Documentation*. This action will create a document labeled **tks2400.doc.** This document describes each function in the instrument. The functions are listed in alphabetical order with a description of the function, C syntax of the function, a description of each parameter, and possible error codes.

SDH for Reference

Following ANSI's development of the SONET standard, the CCITT undertook to define a synchronization standard that would address interworking between the CCITT and ITU transmission hierarchies. That effort culminated in 1989 with CCITT's publication of the Synchronous Digital Hierarchy (SDH) standards. Synchronous Digital Hierarchy is a world standard, and as such, SONET can be considered a subset of SDH.

Transmission standards in the U.S., Canada, Korea, Taiwan, and Hong Kong (ANSI) and the rest of the world (ITU) evolved from different basic-rate signals in the non-synchronous hierarchy. Time Division Multiplexing (TDM) (ANSI) combines twenty four 64-kbps channels (DS0s) into one 1.54-Mb/s DS1 signal. ITU TDM multiplexes thirty-two 64-kbps channels (E0s) into one 2.048 Mb/s E-1 (CEPT format) signal.

The issues between ITU and CCITT standards-makers involved how to efficiently accommodate both the 1.5-Mb/s and the 2-Mb/s non-synchronous hierarchies in a single synchronization standard. The agreement reached specified a basic transmission rate of 52 Mb/s for SONET and a basic rate of 155 Mb/s for SDH.

Synchronous and Non-synchronous line rates and the relationships between each are shown in the two tables that follow.

Table D-1. SONET/SDH Hierarchies

SONET	Bit Rate	SDH	SONET Capacity	SDH Capacity
STS-1, OC-1	51.840 Mb/s	STM-0	28 DS1s or 1 DS3	21 E1s
STS-3, OC-3	155.520 Mb/s	STM-1	84 DS1s or 3 DS3s	63 E1s or 1 E4
STS-12, OC-12	622.080 Mb/s	STM-4	336 DS1s or 12 DS3s	252 E1s or 4 E4s
STS-48, OC-48	2488.320 Mb/s	STM-16	1344 DS1s or 48 DS3s	1008 E1s or 16 E4s
STS-192, OC-192	9953.280 Mb/s	STM-64	5376 DS1s or 192 DS3s	4032 E1s or 64 E4s

It must be noted that although an SDH STM-1 has the same bit rate as the SONET STS-3, the two signals contain different frame structures.

STM = Synchronous Transport Module (ITU)

STS = Synchronous Transfer Signal (ANSI)

OC = Optical Carrier (ANSI)

Table D-2. Non-Synchronous Hierarchies

ANSI Rate			*ITU Rate*		
Signal	**Digital Bit Rate**	**Channels**	**Signal**	**Digital Bit Rate**	**Channels**
DS0	64 kbps	1 DS0	64-kbps	64 kbps	1 64-kbps
DS1	1.544 Mb/s	24 DS0s	E1	2.048 Mb/s	1 E1
DS2	6.312 Mb/s	96 DS0s	E2	8.45 Mb/s	4 E1s
DS3	44.7 Mb/s	28 DS1s	E3	34 Mb/s	16 E1s
not defined			E4	144 Mb/s	64 E1s

Convergence of SONET and SDH Hierarchies

SONET and SDH converge at SONET's 52-Mb/s base level, defined as STM-0 or "Synchronous Transport Module-0". The base level for SONET is STS-1 (or OC-1). Thus, SDH's STM-1 is equivalent to SONET's STS-3 (3 x 51.84 Mb/s = 155.5 Mb/s). Higher SDH rates of STM-4 (622 Mb/s), STM-16 (2.5 Gbps), and STM-64 (10 Gbps) have also been defined.

Multiplexing is accomplished by combining - or interleaving - multiple lower-order signals (1.5 Mb/s, 2 Mb/s, etc.) into higher-speed circuits (52 Mb/s, 155 Mb/s, etc.). By changing the SONET standard from bit-interleaving to byte-interleaving, it became possible for SDH to accommodate both transmission hierarchies. This modification allows an STM-1 signal to carry multiple 1.5-Mb/s or 2-Mb/s signals - and multiple STM signals to be aggregated to carry higher orders of SONET or SDH tributaries.

Asynchronous and Synchronous Tributaries

SDH does away with a number of the lower multiplexing levels, allowing non-synchronous 2-Mb/s tributaries to be multiplexed to the STM-1 level in a single step. SDH recommendations define methods of subdividing the payload area of an STM-1 frame in various ways so that it can carry combinations of synchronous and asynchronous tributaries. Using this method, synchronous transmission systems can accommodate signals generated by equipment operating from various levels of the non-synchronous hierarchy.

SONET for Reference

Transmission standards in the U.S., Canada, Korea, Taiwan, and Hong Kong (ANSI) and the rest of the world (ITU-T) evolved from different basic-rate signals in the non-synchronous hierarchy. ANSI Time Division Multiplexing (TDM) combines twenty four 64 kbit/s channels (DS0) into one 1.544-Mbit/s DS-1 signal. ITU TDM multiplexes thirty 64 kbit/s channels (E0) into one 2.048 Mbit/s E1 signal (an extra two channels provide frame alignment and signaling, making thirty-two (32) total).

An important issue for the ITU-T to resolve was how to efficiently accommodate both the 1.5 Mbit/s and the 2 Mbit/s non-synchronous hierarchies in a single network standard. The agreement reached specified a basic transmission rate of 51 Mbit/s for SONET and a basic rate of 155 Mbit/s for SDH.

Synchronous and Non-synchronous line rates and the relationships between each are shown in the two tables that follow.

Table D-1. SONET/SDH Hierarchies

SONET	Bit Rate	SDH	SONET Capacity	SDH Capacity
STS-1, OC-1	51.84 Mbit/s	STM-0	28 DS1 or 1 DS3	21 E1
STS-3, OC-3	155.52 Mbit/s	STM-1	84 DS1 or 3 DS3	63 E1 or 1 E4
STS-12, OC-12	622.08 Mbit/s	STM-4	336 DS1 or 12 DS3	252 E1 or 4 E4
STS-48, OC-48	2488.32 Mbit/s	STM-16	1344 DS1 or 48 DS3	1008 E1 or 16 E4
STS-192, OC-192	9953.28 Mbit/s	STM-64	5376 DS1 or 192 DS3	4032 E1 or 64 E4

Note: Although an SDH STM-1 has the same bit rate as the SONET STS-3, the two signals contain different frame structures.

STM = Synchronous Transport Module (ITU-T)

STS = Synchronous Transport Signal (ANSI)

OC = Optical Carrier (ANSI)

Table D-2. Non-Synchronous Hierarchies

ANSI Rate			*ITU Rate*		
Signal	**Digital Bit Rate**	**Channels**	**Signal**	**Digital Bit Rate**	**Channels**
DS0	64 kbit/s	1 DS0	64 kbit/s	64 kbit/s	64 kbit/s
DS1	1.544 Mbit/s	24 DS0	E1	2.048 Mbit/s	32 E0
DS2	6.312 Mbit/s	96 DS0	E2	8.448 Mbit/s	128 E0
DS3	44.7 Mbit/s	28 DS1	E3	34.368 Mbit/s	16 E1
not defined			E4	139.264 Mbit/s	64 E1

SONET and SDH Hierarchies

SONET and SDH converge at SDH's 155 Mbit/s base level, defined as STM-1 or "Synchronous Transport Module-1." The base level for SONET is STS-1 (or OC-1) and is equivalent to 51.84 Mbit/s. Thus, SDH's STM-1 is equivalent to SONET's STS-3 (3 x 51.84 Mbit/s = 155.5 Mbit/s). Higher SDH rates of STM-4 (622 Mbit/s), STM-16 (2.4 Gbit/s), and STM-64 (10 Gbit/s) have also been defined.

Multiplexing is accomplished by combining - or interleaving - multiple lower-order signals (1.5 Mbit/s, 2 Mbit/s, etc.) into higher-speed circuits (51 Mbit/s, 155 Mbit/s, etc.). By changing the SONET standard from bit-interleaving to byte-interleaving, it became possible for SDH to accommodate both transmission hierarchies. This modification allows an STM-1 signal to carry multiple 1.5 Mbit/s or 2 Mbit/s signals - and multiple STM signals to be aggregated to carry higher orders of SONET or SDH tributaries.

SDH Glossary & ITU-T Standards

Three sets of terms are arranged at the beginning of this section because it is important to see that they are not equivalent terms, which is how they get used in normal discussions. They are:

Add/ Drop
The process where a part of the information carried in a transmission system is extracted (dropped) at an intermediate point and different information is inserted (added) for subsequent transmission. The remaining traffic passes straight through the multiplexer without additional processing.

Map/ Demap
A term for multiplexing, implying more visibility inside the resultant multiplexed bit stream than available with conventional asynchronous techniques.

Multiplex/ Demultiplex
Multiplex (MUX) - To transmit two or more signals over a single channel. Demultiplex (DEMUX) - To separate two or more signals previously combined by compatible multiplexing equipment. Demultiplexing - A process applied to a multiplex signal for recovering signals combined within it and for restoring the distinct individual channels of the signals.

An alphabetical list of glossary terms follows.

Add/Drop Multiplexer (ADM)
A multiplexer capable of extracting and inserting lower-rate signals from a higher-rate multiplexed signal without completely demultiplexing the signal.

Administrative Unit (AU)
An Administrative Unit is the information structure which provides adaptation between the higher order path layer and the Multiplex Section layer. The Virtual Container (VC) plus the pointers (H1, H2, H3 bytes) is called the Administrative Unit (AU).

AIS (Alarm Indicating Signal)
A code sent downstream indicating an upstream failure has occurred.

AMI
Alternate Mark Inversion. The line-coding format in transmission systems where successive ones (marks) are alternatively inverted (sent with polarity opposite that of the preceding mark).

Analog Jitter Out
A signal that contains the demodulated jitter from a line or clock input.

ANSI
American National Standards Institute. A standards-setting, non-government organization, which develops and publishes standards for ”voluntary” use in the United States.

Asynchronous
A network where transmission system payloads are not synchronized and each network terminal runs on its own clock.

Asynchronous Transfer Mode (ATM)
A multiplexing/switching technique in which information is organized into fixed-length cells with each cell consisting of an identification header field and an information field. The transfer mode is asynchronous in the sense that the use of the cells depends on the required or instantaneous bit rate.

Attenuation
Reduction of signal magnitude or signal loss, usually expressed in decibels.

AU–4
Virtual Container (VC) plus the Transport Overhead pointers.

AU-N
Administrative Unit-N; a discrete unit of the SDH payload carrying one or more VC-N

Automatic Protection Switching (APS)
The ability of a network element to detect a failed working line and switch the service to a spare (protection) line. 1+1 APS pairs a protection line with each working line. 1:N APS provides one protection line for every N working lines.

Backhauling
Cumbersome traffic management technique used to reduce the expense of multiplexing/demultiplexing.

Bandwidth
Information-carrying capacity of a communication channel. Analog bandwidth is the range of signal frequencies that can be transmitted by a communication channel or network.

Bidirectional
Operating in both directions. Bi-directional APS allows protection switching to be initiated by either end of the line.

Binary N-Zero Suppression (BNZS)
Line coding system that replaces N number of zeros with a special code to maintain pulse density required for clock recovery. N is typically 3, 6, or 8.

BIP-8 (Bit Interleaved Parity-8)
A method of error checking in SDH which allows in-service performance monitoring. For example, a BIP-8 creates eight-bit (one-byte) groups, then does a parity check for each of the eight bit positions in the byte.

B-ISDN (Broadband Integrated Services Digital Network)
A single ISDN network which can handle voice, data, and eventually video services.

Bit
One binary digit; a pulse of data.

Bit Error vs. Block Error

Error rate statistics play a key role in measuring the performance of a network. As errors increase, user payload (especially data) must be re-transmitted, or lost entirely.

Bit Error Rate (BER) - The number of bit errors detected in a unit of time, usually one second. Bit Error rate (BER) is calculated with this formula:

BER = errored bits received/ total bits sent.

Block Error Rate (BLER) - One of the underlying concepts of error performance is the notion of Errored Blocks, i.e., blocks in which one or more bits are in error. A block is a set of consecutive bits associated with the path or section monitored by means of an Error Detection Code (EDC), such as Bit Interleaved Parity (BIP). Block Error rate (BLER) is calculated with this formula:

BLER = errored blocks received/ total blocks sent.

Bit-Interleaved Parity (BIP)
A parity check that groups all the bits in a block into units (such as byte), then performs a parity check for each bit position in the group.

Bit-Stuffing
In asynchronous systems, a technique used to synchronize asynchronous signals to a common rate before multiplexing.

Bits per second (bit/s)
The number of bits passing a point every second. The transmission rate for digital information.

Broadband
Services requiring over 2 Mbit/s transport capacity.

CCITT
See ITU.

Channel
The smallest subdivision of a circuit that provides a type of communication service; usually a path with only one direction.

Circuit
A communications path or network; usually a pair of channels providing bi-directional communication.

Circuit Switching
Basic switching process whereby a circuit between two users is opened on demand and maintained for their exclusive use for the duration of the transmission.

Coding Violation (CV)
A transmission error detected by the difference between the transmitted line code and that expected at the receive end by the logical coding rules.

Concatenation
The linking together of various data structures, for example two channels joined to form a single channel. In SDH, a number (M) of TUs can be linked together to produce a concatenated container, M times the size of the TU. An example of this is the concatenation of five TU-2s to carry a 32 Mbit/s video signal, known as VC-2-5c. Once assembled, any concatenated VC structure is multiplexed, switched and transported through the network as a single entity.

Conformance Tests
Conformance Tests (also know as Standards Conformance Tests) are predefined tests that measure the impact of jitter or wander on networks or network elements. They are defined in the ITU–T Recommendations.

Cyclic Redundancy Check (CRC)
A technique for using overhead bits to detect transmission errors.

Data Communications Channel (DCC)
Data channels in SDH that enable OAM communications between intelligent controllers and individual network nodes as well as inter-node communications.

dB
The symbol for decibels.

dBm
The symbol for power level in decibels relative to 1 mW.

Defect
A limited interruption in the ability of an item to perform a required function. Persistence of a defect can cause a failure.

Digital Cross-connect (DCS)
An electronic cross-connect which has access to lower-rate channels in higher-

rate multiplexed signals and can electronically rearrange (cross-connect) those channels.

Digital Signal
An electrical or optical signal that varies in discrete steps. Electrical signals are coded as voltages, optical signals are coded as pulses of light.

E1, E2, E3, E4
Alternative names for the ITU-T 2 Mb/s, 8 Mb/s, 32 Mb/s, and 140 Mb/s tributary signals.

ES
Errored Second; measure of network or equipment performance

ETSI (European Telecommunications Standards Institute)
Organization responsible for defining and maintaining European standards, including SDH.

Failure
A termination of the ability of an item to perform a required function. A failure is caused by the persistence of a defect.

FEBE (Far End Block Error)
See Remote Error Indication (REI).

FERF (Far End Receive Failure)
See Remote Defect Indication (RDI).

FIFO
First-In First-Out; a type of data buffer

Fixed Stuff
A bit or byte whose function is reserved. Fixed stuff locations, sometimes called reserved locations, do not carry overhead or payload.

Floating mode
A virtual tributary mode that allows the VC synchronous payload envelope to begin anywhere in the VC. Pointers identify the starting location of the LO-VC. LO-VCs in different multiframes may begin at different locations.

Framing
Method of distinguishing digital channels that have been multiplexed together.

Frequency
The number of cycles of periodic activity that occur in a discrete amount of time.

Frequency Drift Rate
The rate of change of a line or clock frequency, measured in ppm/sec. A new measure of tining quality.

Fullband
A jitter filter range that includes the wideband range but extends the low–frequency cutoff to 10 Hz or below. This filter is useful when analyzing video timing quality.

Grooming
Consolidating or segregating traffic for efficiency.

HDB3
High Density Bipolar 3. A bipolar coding method that does not allow more than three consecutive zeros.

Highband
A jitter filter range that measures high–frequency jitter. The band pass filters are defined in ITU–T standards (where they are sometimes referred to as HP2 - LP1).

Highpass
The lower -3 dB corner frequency of a filter. The filter passes frequencies higher than this frequency.

Interleave
The ability of SDH to mix together and transport different types of input signals in an efficient manner, thus allowing higher-transmission rates.

ITU (International Telecommunications Union)
An agency of the United Nations responsible for the regulation, standardization, co-ordination and development of international telecommunications as well as the harmonization of national policies. It functions through international committees of telecommunications administrations, operators, manufacturers and scientific/industrial organizations.

Jitter
The short term variations of the significant instants of a timing signal from their ideal positions in time (where short term implies that these variations are of frequency greater than or equal to 10 Hz).

Jitter Hit
When peak–to–peak jitter crosses a predetermined threshold.

Jitter Tolerance
A conformance test that measures the susceptability of a network element input to incoming jitter.

Jitter Transfer
A conformance test that measures the transfer of jitter from the input to the output of a network element.

Jitter Transfer Function
A graph that shows jitter gain as a function of jitter frequency.

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 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.

Locked Mode
A virtual tributary mode that fixes the starting location of the VC. Locked mode has less pointer processing than floating mode.

LOF
An acronym for Loss of Frame.

LOP
An acronym for Loss of Pointer.

LOS
An acronym for Loss of Signal.

Low Frequency Jitter
Jitter that crosses the wander threshold (approximately 0.1 Hz to 500 Hz). Low frequency jitter is often the result of pointer movement.

Lowpass
The upper -3 dB corner frequency of a filter. The filter passes frequencies lower than this frequency.

LTE
An acronym for Line Terminating Equipment.

Mapping
The process of associating each bit transmitted by a service into the SDH payload structure that carries the service. For example, mapping a E1 service into a SDH VC-12 associates each bit of the E1 with a location in the VC-12.

Multiframe
Any structure made of multiple frames. SDH has facilities to recognize multiframes at the E1 level and at the VC-n level.

Multiplex Section Alarm Indication Signal (MS-AIS)
MS-AIS is generated by Section Terminating Equipment (STE) upon the detection of a Loss of Signal or Loss of Frame defect, on an equipment failure. MS-AIS maintains operation of the downstream regenerators, and therefore prevents generation of unnecessary alarms. At the same time, data and orderwire communication is retained with the downstream Line Terminating Equipment (LTE).

Multiplex Section Remote Defect Indication (MS-RDI)
A signal returned to the transmitting Line Terminating Equipment (LTE) upon detecting a Loss of Signal, Loss of Frame, or MS-AIS defect. MS-RDI was previously known as Multiplex Section FERF.

Multiplex Section Overhead (MSOH)
18 bytes of overhead accessed, generated, and processed by MS terminating equipment. This overhead supports functions such as locating the payload in the frame, multiplexing or concatenating signals, performance monitoring, automatic protection switching and line maintenance.

Multiplexer
A device for combining several channels to be carried by a single physical channel.

Narrowband
Services requiring up to 2-Mbit/s transport capacity.

Network Element (NE) - In SDH, the five basic network elements are:
add/drop multiplexer;
broadband digital cross-connect;
wideband digital cross-connect;
flexible multiplexer; and,
regenerator.

Any device which is part of a SDH transmission path and serves one or more of the section, line and path-terminating functions.

OAM
Operations, Administration, and Maintenance. Also called OAM&P.

OAM&P (Operations, Administration, Maintenance, and Provisioning)
Provides the facilities and personnel required to manage a network.

Orderwire
A dedicated voice channel used by installers to expedite the provisioning of lines.

OOF
An acronym for Out of Frame.

OS (Operations System)
Sophisticated applications software that manages operation of the entire network.

OSI Seven-layer Model
A standard architecture for data communications. Layers define hardware and software required for multi-vendor information processing equipment to be mutually compatible. The seven layers from lowest to highest are: physical, link, network, transport, session, presentation, and application.

Output Jitter
A compliance test the measures the output jitter of a network or network element.

Overhead
Extra bits in a digital stream used to carry information besides traffic signals. Orderwire, for example, would be considered overhead information.

Packet Switching
An efficient method for breaking down and handling high-volume traffic in a network. A transmission technique that segments and routes information into discrete units. Packet switching allows for efficient sharing of network resources as packets from different sources can all be sent over the same channel in the same bitstream.

Parity check
An error-checking scheme which examines the number of transmitted bits in a block which hold the value of ”one”. For even parity, an overhead parity bit is set to either one or zero to make the total number of transmitted ones in the data block plus parity bit an even number. For odd parity, the parity bit is set to make the total number of ones in the block an odd number.

Path
A logical connection between a point where a service in a VC is multiplexed to the point where it is demultiplexed.

Path Overhead (POH)
Overhead accessed, generated, and processed by path-terminating equipment.

Path Terminating Equipment (PTE)
Network elements such as fibre optic terminating systems which can access, generate, and process Path Overhead.

Payload
The portion of the SDH signal available to carry service signals such as E1 and E3. The contents of a VC.

Payload Pointer
Indicates the beginning of a Virtual Container.

Payload capacity
The number of bytes the payload of a single frame can carry.

Plesiochronous
A network with nodes timed by separate clock sources with almost the same timing.

PLL
Phase-Locked Loop; method of timing recovery

Pointer
A part of the SDH overhead that locates a floating payload structure. AU-n pointers locate the payload. TU-m Pointers locate floating mode virtual tributaries. All SDH frames use AU pointers; only floating mode virtual containers use TU pointers.

Pointer Hit
When the line or clock frequency drift crosses a predetermined threshold.

Pointer Jitter
A measure of the jitter output of a network when specified pointer test sequences are applied to its input.

ppTIE
Peak-to-Peak Time Interval Error , a measure of wander

PRC (Primary Reference Clock)
In a synchronous network, all the clocks are traceable to one highly stable reference supply, the Primary Reference Clock (PRC). The accuracy of the PRC is better than ±1 in 10^{11} and is derived from a cesium atomic standard.

Remote Alarm Indication (RAI)
A code sent upstream in a E-*n* network as a notification that a failure condition has been declared downstream. (RAI signals were previously referred to as Yellow signals.)

Remote Defect Indication (RDI)
A signal returned to the transmitting Terminating Equipment when the receiving Terminating Equipment detects a Loss of Signal, Loss of Frame, or AIS defect. RDI was previously known as FERF.

Remote Error Indication (REI)
An indication returned to a transmitting node (source) that an errored block has been detected at the receiving node (sink). This indication was formerly known as Far End Block Error (FEBE).

Remote Failure Indication (RFI)
A failure is a defect that persists beyond the maximum time allocated to the transmission system protection mechanisms. When this situation occurs, an RFI is sent to the far end and will initiate a protection switch if this function has been enabled.

Regenerator
Device that restores a degraded digital signal for continued transmission; also called a repeater.

rms
Root Mean Square; calculation often applied to power and noise measurements

SDH (Synchronous Digital Hierarchy)
The ITU-defined international networking standard whose base transmission level is 155 Mbit/s (STM-1). SDH standards were first published in 1989 to address interworking between the ITU and ANSI transmission hierarchies.

SEC (Synchronous Equipment Clock)
G.813 slave clock contained within a SDH network element.

Section
The span between two SDH network elements capable of accessing, generating, and processing only SDH Section overhead.

Section Overhead
Nine columns of overhead accessed, generated, and processed by section terminating equipment. This overhead supports functions such as framing the signal and performance monitoring.

Section Terminating Equipment (STE)
Equipment that terminates the SDH Section layer. STE interprets and modifies or creates the Section Overhead.

SES
Severely Errored Second; measure of network performance

Slip
An overflow (deletion) or underflow (repetition) of one frame of a signal in a receiving buffer.

SONET (Synchronous Optical Network)
A standard for optical transport in the United, States, Canada, Japan, Korea and Hong Kong that defines optical carrier levels and their electrically equivalent synchronous transport signals. SONET allows for a multi-vendor environment and positions the network for transport of new services, synchronous networking, and enhanced OAM&P.

SSM (Synchronisation Status Message)
Bits 5 to 8 of SDH overhead byte S1 are allocated for Synchronisation Status Messages. See further details on the assignment of bit patterns for byte S1 in the section of this primer on Multiplex Section Overhead.

Stuffing
see bit-stuffing

Synchronous
A network where transmission system payloads are synchronized to a master (network) clock and traced to a reference clock. A network where all clocks have the same long term accuracy under normal operating conditions.

Synchronous Equipment Timing Source (SETS)
A network equipment clock.

Synchronous Transport Module (STM)
A structure in the SDH transmission hierarchy. STM-1 is SDH’s base-level transmission rate equal to 155 Mbit/s. Higher rates of STM-4, STM-16, and STM-64 are also defined.

TDEV
Time Deviation; a measure of wander

Through Mode
The ability to retransmit the incoming signal and manipulate its contents.

TIE
Time Interval Error is the time difference in nanoseconds between the nominal value of a line or clock period and the actual received period.

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.

Tributary Unit (TU)
A Tributary Unit is an information structure which provides adaptation between the lower order path layer and the higher order path layer. It contains the Virtual Container (VC) plus a tributary unit pointer.

Tributary Unit Group (TUG)
Contains several Tributary Units.

UI
Unit Interval; a measure of jitter

UIpp
Unit Interval Peak-to-Peak; a common measure of jitter

UIrms
Unit Interval rms; a measure of jitter in line systems

Virtual Container (VC)
A signal designed for transport and switching of sub-SDH payloads.

Wander
The long term variations of the significant instants of a digital signal from their ideal position in time (where long term implies that these variations are of frequency less than 10 Hz).

Wideband
A jitter filter range that measures jitter over a wide range. The band pass filters are defined in ITU–T standards (where they are sometimes referred to as HP1 - LP1).

Yellow Signal
See Remote Alarm Indication (RAI).

SDH Standards for ITU-T

G.701
Vocabulary of digital transmission and multiplexing and PCM terms

G.702
Digital Hierarchy bit rates

G.703
Physical/ electrical characteristics of hierarchical digital interfaces

G.704
Synchronous frame structures used at 1544, 6312, 2048, 8448 and 4476 kbit/s hierarchical levels

G.706
Frame alignment and cyclic redundancy check (CRC) procedures relating to basic frame structures defined in Recommendation G.704

G.707
Network Node Interface for the SDH

G.772
Protected monitoring points provided on digital transmission systems

G.780
Vocabulary of terms for SDH networks and equipment

G.782
Types and general characteristics of SDH equipment

G.783
Characteristics of SDH equipment functional blocks

G.784
SDH management

G.802
Interworking between networks based on different and speech coding laws

G.803
Architectures of transport networks based on the SDH

G.825
Control of jitter and wander in SDH

G.826
Error performance parameters and objectives for international, constant bit rate digital paths at or above the primary rate

G.831
Management capabilities of transport network based on SDH

G.841
Types and characteristics of SDH network protection architectures

G.861
Principles and guidelines for the integration of satellite and radio systems in SDH

G.957
Optical interfaces for equipment and systems relating to the SDH

G.958
Digital line systems based on SDH for use on optical fibre cables

I.432
B-ISDN User -network interface - Physical layer specification

M.2101
Performance Limits for BIS and Maintenance of Int’ SDH Paths and Multiplex Sections

O.150
General requirements for instrumentation for performance measurements on digital transmission equipment

O.181
Equipment to assess error performance on STM-N interfaces

F.750 (ITU-R)
Architectures and functional aspects of radio-relay systems for SDH-based networks

SONET Glossary and Standards

Three sets of terms are arranged at the beginning of this section because it is important to see that they are not equivalent terms, which is how they get used in normal discussions. They are:

- **Add/ Drop**

The process where a part of the information carried in a transmission system is demodulated (dropped) at an intermediate point and different information is entered (added) for subsequent transmission. The remaining traffic passes straight through the multiplexer without additional processing.

- **Map/ Demap**

A term for multiplexing, implying more visibility inside the resultant multiplexed bit stream than available with conventional asynchronous techniques.

- **Multiplex/ Demultiplex**

Multiplex (MUX) - To transmit two or more signals over a single channel. Demultiplex (DEMUX) - To separate two or more signals previously combined by compatible multiplexing equipment. Demultiplexing - A process applied to a multiplex signal for recovering signals combined within it and for restoring the distinct individual channels of the signals.

An alphabetical list of glossary terms follows.

Add/Drop Multiplexer (ADM) - A multiplexer capable of extracting and inserting lower-rate signals from a higher-rate multiplexed signal without completely demultiplexing the signal.

AIS (Alarm Indicating Signal) - A code sent downstream indicating an upstream failure has occurred. SONET defines four categories of AIS: Line AIS; STS Path AIS; VT Path AIS; DSn AIS.

AMI - Alternate Mark Inversion. The line-coding format in transmission systems where successive ones (marks) are alternatively inverted (sent with polarity opposite that of the preceding mark).

ANSI (American National Standards Institute) - A membership organization which develops U.S. industry standards and coordinates U.S. participation in the International Standards Organization (ISO).

Asynchronous - A network where transmission system payloads are not synchronized and each network terminal runs on its own clock.

Asynchronous Transfer Mode (ATM) - A multiplexing/switching technique in which information is organized into fixed-length cells with each cell consisting of an identification header field and an information field. The transfer mode is asynchronous in the sense that the use of the cells depends on the required or instantaneous bit rate.

Attenuation - Reduction of signal magnitude or signal loss, usually expressed in decibels.

Automatic Protection Switching (APS) - The ability of a network element to detect a failed working line and switch the service to a spare (protection) line. 1+1 APS pairs the protection line with each working line. 1:*n* APS provides one protection line for every *n* working lines.

Backhauling - Cumbersome traffic management technique used to reduce expense of multiplexing/demultiplexing.

Bandwidth - Information-carrying capacity of a communication channel. Analog bandwidth is the range of signal frequencies that can be transmitted by a communication channel or network.

Bidirectional - Operating in both directions. Bidirectional APS allows protection switching to be initiated by either end of the line.

Binary N-Zero Suppression (BNZS) - Line coding system that replaces N number of zeros with a special code to maintain pulse density required for synchronization. N is typically 3, 6, or 8.

BIP-8 (Bit Interleaved Parity-8) - A method of error checking in SONET which allows a full set of performance statistics to be generated. For example, a BIP-8 creates eight-bit (one-byte) groups, then does a parity check for each of the eight bit positions in the byte.

BISDN (Broadband Integrated Services Digital Network) - A single ISDN network which can handle voice, data, and eventually video services.

Bit - One binary digit; a pulse of data.

Bit Error vs. Block Error

Error rate statistics play a key role in measuring the performance of a network. As errors increase, user payload (especially data) must be re-transmitted. The end effect is creation of more (non-revenue) traffic in the network.

Bit Error Rate (BER) - The number of coding violations detected in a unit of time, usually one second. Bit Error rate (BER) is calculated with this formula:

BER = errored bits received/ total bits sent.

Block Error rate (BLER) - One of the underlying concepts of error performance is the notion of Errored Blocks, for examples blocks in which one or more bits are in error. A block is a set of consecutive bits associated with the path or section monitored by means of an Error Detection Code (EDC), such as Bit Interleaved Parity (BIP). Block Error rate (BLER) is calculated with this formula:

BLER = errored blocks received/ total block sent.

Bit-Interleaved Parity (BIP) - A parity check that groups all the bits in a block into units (such as byte), then performs a parity check for each bit position in a group.

Bit Stuffing - In asynchronous systems, a technique used to synchronize asynchronous signals to a common rate before multiplexing.

Bit synchronous - A way of mapping payload into virtual tributaries that synchronizes all inputs into the VTs, but does not capture any framing information or allow access to subrate channels carried in each input. For example, bit synchronous mapping of a channeled DS1 into a VT1.5 does not provide access to the DS0 channels carried by the DS1.

Bits per second (bps) - The number of bits passing a point every second. The transmission rate for digital information.

Broadband - Services requiring 50-600 Mb/s transport capacity.

Byte-interleaved - Bytes from each STS-1 are placed in sequence in a multiplexed or concatenated STS-N signal. For example, for an STS-3, the sequence of bytes from contributing STS-1s is 1,2,3,1,2,3,...

Byte synchronous - A way of mapping payload into virtual tributaries that synchronizes all inputs into the VTs, captures framing information, and allows access to subrate channels carried in each input. For example, byte synchronous mapping of a channeled DS1 into a VT1.5 provides direct access to the DS0 channels carried by the DS1.

CCITT - The technical organs of the United Nations specialized agency for telecommunications, now the International Telecommunications Union. They function through international committees of telephone administrations and private operating agencies.

CEPT - European Conference of Postal and Telecommunications Administrations. The CEPT format defines the 2.048-Mb/s European T1/E1 signal made up of 32 voice-frequency channels.

Channel - The smallest subdivision of a circuit that provides a type of communication service; usually a path with only one direction.

Circuit - A communications path or network; usually a pair of channels providing bi-directional communication.

Circuit Switching - Basic switching process whereby a circuit between two users is opened on demand and maintained for their exclusive for the duration of the transmission.

Coding Violation (CV) - A transmission error detected by the difference between the transmitted and the locally calculated bit-interleaved parity.

Concatenate - The linking together of various data structures, for example two bandwidths joined to form a single bandwidth.

Concatenated STS-Nc - A signal in which the STS Envelope Capacities from the N STS-1s have been combined to carry an STS-Nc Synchronous Payload Envelope (SPE). It is used to transport signals that do not fit into an STS-1 (52 Mb/s) payload.

Concatenated VT - A virtual tributary (VT x Nc) which is composed of N x VTs combined. Its payload is transported as a single entity rather than separate signals.

Cyclic Redundancy Check (CRC) - A technique for using overhead bits to detect transmission errors.

Data Communications Channels - OAM&P channels in SONET that enable communications between intelligent controllers and individual network nodes as well as inter-node communications.

Defect - A limited interruption in the ability of an item to perform a required function.

Digital Cross-Connect (DCS) - An electronic cross-connect which has access to lower-rate channels in higher-rate multiplexed signals and can electronically rearrange (cross-connect) those channels.

Digital Signal - An electrical or optical signal that varies in discrete steps. Electrical signals are coded as voltages, optical signals are coded as pulses of light.

DSX-1 - May refer to either a cross-connect for DS1 rate signals or the signals cross-connected at an DSX-1.

DSX-3 - May refer to either a cross-connect for DS3 rate signals or the signals cross-connected at an DSX-1.

ECSA (Exchange Carrier Standards Association) - An organization that specifies telecommunications standards for ANSI.

Envelope capacity - The number of bytes the payload envelope of a single frame can carry. The SONET STS payload envelope is the 783 bytes of the STS-1 frame available to carry a signal. Each virtual tributary has an envelope capacity defined as the number of bytes in the virtual tributary less the bytes used by VT overhead.

Failure - A termination of the ability of an item to perform a required function. A failure is cause by the persistence of a defect.

FEBE (Far End Block Error) - A message sent back upstream that receiving network element is detecting errors, usually a coding violation. See Remote Error Indication (REI).

FERF (Far End Receive Failure) - A signal to indicate to the transmit site that a failure has occurred at the receive site.

Fixed Stuff - A bit or byte whose function is reserved. Fixed stuff locations, sometimes called reserved locations, do not carry overhead or payload.

Floating mode - A virtual tributary mode that allows the VT synchronous payload envelope to begin anywhere in the VT. Pointers identify the starting location of the VT SPE. VT SPEs in different superframes may begin at different locations.

Framing - Method of distinguishing digital channels that have been multiplexed together.

Frequency - The number of cycles of periodic activity that occur in a discrete amount of time.

Grooming - Consolidating or segregating traffic for efficiency.

Interleave - The ability of SONET to mix together and transport different types of input signals in an efficient manner, thus allowing higher-transmission rates.

Isochronous - All devices in the network derive their timing signal directly or indirectly from the same primary reference clock.

Jitter - Short waveform variations caused by vibration, voltage fluctuations, control system instability, etc.

Line - One or more SONET sections, including network elements at each end, capable of accessing, generating, and processing Line Overhead.

Line Alarm Indication Signal (AIS-L) - AIS-L is generated by Section Terminating Equipment (STE) upon the detection of an Loss of Signal or Loss of Frame defect, on an equipment failure. AIS-L maintains operation of the downstream regenerators, and therefore prevents generation of unnecessary alarms. At the same time, data and orderwire communication is retained between the regenerators and the downstream Line Terminating Equipment (LTE).

Line Remote Defect Indication (RDI-L) - A signal returned to the transmitting Line Terminating Equipment (LTE) upon detecting a Loss of Signal, Loss of Frame, or AIS-L defect. RDI-L was previously known as Line FERF.

Line Overhead (LOH) - 18 bytes of overhead accessed, generated, and processed by line terminating equipment. This overhead supports functions such as locating the SPE in the frame, multiplexing or concatenating signals, performance monitoring, automatic protection switching and line maintenance.

Line Terminating Equipment (LTE) - Network elements such as add/drop multiplexers or digital cross-connect systems which can access, generate, and process Line Overhead.

Locked Mode - A virtual tributary mode that fixes the starting location of the VT SPE. Locked mode has less pointer processing than floating mode.

Mapping - The process of associating each bit transmitted by a service into the SONET payload structure that carries the service. For example, mapping a DS1 service into a SONET VT1.5 associates each bit of the DS1 with a location in the VT1.5.

Multiplexer - A device for combining several channels to be carried by one line or fiber.

Mesochronous - A network whereby all nodes are timed to a single clock source, thus all timing is exactly the same (truly synchronous).

Narrowband - Services requiring up to 1.5 Mb/s transport capacity.

Network Element (NE) - In SONET, the five basic network elements are:

add/drop multiplexer;
broadband digital cross-connect;
wideband digital cross-connect;
digital loop carrier; and ,
switch interface.

Any device which is part of a SONET transmission path and serves one or more of the section, line and path-terminating functions.

OAM - Operations, Administration, and Maintenance. Also called OAM&P.

OAM&P (Operations, Administration, Maintenance, and Provisioning) - Provides the facilities and personnel required to manage a network.

OC-1 (Optical Carrier Level 1) - The optical equivalent of an STS-1 signal.

OC-n (Optical Carrier Level n) - The optical equivalent of an STS-n signal.

Orderwire - A channel used by installers to expedite the provisioning of lines.

OS (Operations System) - Sophisticated applications software that overlooks the entire network.

OSI Seven-layer Model - A standard architecture for data communications. Layers define hardware and software required for multi-vendor information processing equipment to be mutually compatible. The seven layers from lowest to highest are: physical, link, network, transport, session, presentation, and application.

Overhead - Extra bits in a digital stream used to carry information besides traffic signals. Orderwire, for example, would be considered overhead information.

Packet Switching - An efficient method for breaking down and handling high-volume traffic in a network. A transmission technique that segments and routes information into discrete units. Packet switching allows for efficient sharing of network resources as packets from different sources can all be sent over the same channel in the same bitstream.

Parity check - An error-checking scheme which examines the number of transmitted bits in a block which hold the value one. For even parity, an overhead parity bit is set to either one or zero to make the total number of transmitted one in the block data plus parity bit an even number. For odd parity, the parity bit is set to make the total number of ones in the block an odd number.

Path - A logical connection between a point where an STS or VT is multiplexed to the point where it is demultiplexed.

Path Overhead (POH) - Overhead accessed, generated, and processed by path-terminating equipment. Path overhead includes nine bytes of STS Path Overhead and, when the frame is VT-structured, five bytes of VT Path Overhead.

Path Terminating Equipment (PTE) - Network elements such as fiber optic terminating systems which can access, generate, and process Path Overhead.

Payload - The portion of the SONET signal available to carry service signals such as DS1, DS2, and DS3. The contents of an STS SPE or VT SPE.

Payload Pointer - Indicates the beginning of the Synchronous Payload Envelope.

Photonic - The basic unit of light transmission used to define the lowest (physical) layer in the OSI seven-layer model.

Plesiochronous - A network with nodes timed by separate clock sources with almost the same timing.

Pointer - A part of the SONET overhead that locates a floating payload structure. STS pointers locate the SPE. VT Pointers locate floating mode virtual tributaries. All SONET frames use STS pointers; only floating mode virtual tributaries use VT pointers.

Poll - An individual control message from a central controller to an individual station on a multipoint network inviting that station to send.

POP (Point-of-Presence) - A point in the network where inter-exchange carrier facilities like DS3 or OC-n meet with access facilities managed by telephone companies or other service providers.

Remote Alarm Indication (RAI) - A code sent upstream in a DSn network as a notification that a failure condition has been declared downstream. (RAI signals were previously referred to as Yellow signals.)

Remote Defect Indication (RDI) - A signal returned to the transmitting Terminating Equipment upon detecting a Loss of Signal, Loss of Frame, or AIS defect. RDI was previously known as FERF.

Remote Error Indication (REI) - An indication returned to a transmitting node (source) that an errored block has been detected at the receiving node (sink). This indication was formerly known as Far End Block Error (FEBE).

Remote Failure Indication (RFI) - A failure is a defect that persists beyond the maximum time allocated to the transmission system protection mechanisms. When this situation occurs, an RFI is sent to the far end and will initiate a protection switch if this function has been enabled.

Regenerator - Device that restores a degraded digital signal for continued transmission; also called a repeater.

SDH (Synchronous Digital Hierarchy) - The CCITT-defined world standard of synchronization whose base transmission level is 52 Mb/s (STM-0) and is equivalent to SONET's STS-1 or OC-1 transmission rate. SDH standards were published in 1989 to address interworking between the CCITT (ITU) and ANSI transmission hierarchies.

Section - The span between two SONET network elements capable of accessing, generating, and processing only SONET Section overhead. This is the lowest layer of the SONET protocol stack with overhead.

Section Overhead - Nine bytes of overhead accessed, generated, and processed by section terminating equipment. This overhead supports functions such as framing the signal and performance monitoring.

Section Terminating Equipment (STE) - Equipment that terminates the SONET Section layer. STE interprets and modifies or creates the Section Overhead.

Slip - An overflow (deletion) or underflow (repetition) of one frame of a signal in a receiving buffer.

SONET (Synchronous Optical Network) - A standard for optical transport that defines optical carrier levels and their electrically equivalent synchronous transport signals. SONET allows for a multi-vendor environment and positions the network for transport of new services, synchronous networking, and enhanced OAM&P.

Stratum - Level of clock source used to categorize accuracy.

Superframe - Any structure made of multiple frames. SONET recognizes superframes at the DS1 level (D4 and extended superframe) and at the VT (500u STS superframes).

Synchronous - A network where transmission system payloads are synchronized to a master (network) clock and traced to a reference clock.

Synchronous Transfer Module (STM) - A measure of the SDH transmission hierarchy. STM-1 is SDH's base-level transmission rate equal to 155 Mb/s. Higher rates of STM-4, STM-16, and STM-48 are also defined.

SPE (Synchronous Payload Envelope) - The major portion of the SONET frame format used to transport payload and STS path overhead. A SONET structure that carries the payload (service) in a SONET frame or virtual tributary. The STS SPE may begin anywhere in the frame's payload envelope. The VT SPE may begin anywhere in a floating mode VT, but begins at a fixed location in a locked-mode VT.

STS Path Overhead (STS POH) - Nine evenly distributed Path Overhead bytes per 125 microseconds starting at the first byte of the STS SPE. STS POH provides for communication between the point of creation of an STS SPE and its point of disassembly.

STS Path Remote Defect Indication (RDI-P) - A signal returned to the transmitting STS Path Terminating Equipment (PTE) upon detection of certain defects on the incoming path.

STS Path Terminating Equipment (STS PTE) - Equipment that terminates the SONET STS Path layer. STS PTE interprets and modifies or creates the STS Path Overhead. An NE that contains STS PTE will also contain LTE and STE.

STS-1 (Synchronous Transport Signal Level 1) - The basic SONET building block signal transmitted at 51.84 Mb/s data rate.

STS-n (Synchronous Transport Signal Level n) - The signal obtained by multiplexing integer multiples (N) of STS-1 signals together.

T1X1 Subcommittee - A committee within the ECSA that specifies SONET optical interface rates and formats.

VT (Virtual Tributary) - A signal designed for transport and switching of sub-STS-1 payloads.

VT Group - A 9 row x 12 column structure (108 bytes) that carries one or more VTs of the same size. Seven VT groups can be fitted into one STS-1 payload.

VT Path Overhead (VT POH) - Four evenly distributed Path Overhead bytes per VT SPE starting at the first byte of the VT SPE. VT POH provides for communication between the point of creation of an VT SPE and its point of disassembly.

VT Path Remote Defect Indication (RDI-V) - A signal returned to the transmitting VT PTE upon detection of certain defects on the incoming path.

VT Path Remote Failure Indication (RFI-V) - A signal, applicable only to a VT1.5 with the byte-synchronous DS1 mapping, that is returned to the transmitting VT PTE upon declaring certain failures. The RFI-V signal was previously known as the VT Path Yellow signal.

VT Path Terminating Equipment (VT PTE) - Equipment that terminates the SONET VT Path layer. VT PTE interprets and modifies or creates the VT Path Overhead. An NE that contains VT PTE will also contain STS PTE, LTE and STE.

Wander - Long-term variations in a waveform.

Wideband - Services requiring 1.5-50 Mb/s transport capacity.

Yellow Signal - See Remote Alarm Indication (REI) and VT Path Remote Failure Indication (RFI-V).

SONET Reference Materials

Bellcore

GR-253-CORE

SONET Transport Systems: Common Generic Criteria

Consult this document for an up-to-date listing of:

Generic Requirements (GR)

Technical References)TR)

Technical Advisories (TA)

Special Reports (SR)

EIA/TIA Documents

American National Standards Institute (ANSI) documents

ITU-T and CCITT Recommendations

ISO documents

IEEE documents