Operation and Service Manual

Optical Interface Controller

SX199

SRS Stanford Research Systems

Revision 1.02 • August 6, 2013

Certification

Stanford Research Systems certifies that this product met its published specifications at the time of shipment.

Warranty

This Stanford Research Systems product is warranted against defects in materials and workmanship for a period of one (1) year from the date of shipment.

Service

For warranty service or repair, this product must be returned to a Stanford Research Systems authorized service facility. Contact Stanford Research Systems or an authorized representative before returning this product for repair.

Information in this document is subject to change without notice.

Stanford Research Systems, Inc.
1290–D Reamwood Avenue
Sunnyvale, CA 94089 USA
Phone: (408) 744-9040 • Fax: (408) 744-9049
www.thinkSRS.com • e-mail: info@thinkSRS.com

Printed in U.S.A. Document number 9-01692-903

Contents

General Information **iii**

Safety and Preparation for Use iii

Symbols iv

Notation v

Specifications vi

1 Introduction **1 – 1**

1.1 Introduction to the instrument . . . . 1 – 2

1.1.1 Front Panel . . . . 1 – 2

1.1.2 Rear Panel . . . . 1 – 3

1.2 Remote configuration . . . . 1 – 3

1.2.1 IEEE-488 GPIB . . . . 1 – 3

1.2.2 Ethernet . . . . 1 – 5

1.2.3 RS-232 . . . . 1 – 6

1.3 Debug port . . . . 1 – 6

2 Remote Operation **2 – 1**

2.1 Index of commands . . . . 2 – 2

2.2 Alphabetic list of commands . . . . 2 – 3

2.3 Introduction . . . . 2 – 5

2.3.1 Interface configuration . . . . 2 – 5

2.3.2 Buffers . . . . 2 – 5

2.4 Ethernet . . . . 2 – 5

2.5 Link model . . . . 2 – 6

2.6 Commands . . . . 2 – 7

2.6.1 Command syntax . . . . 2 – 7

2.6.2 Notation . . . . 2 – 8

2.6.3 Examples . . . . 2 – 8

2.6.4 Link commands . . . . 2 – 9

2.6.5 Interface commands . . . . 2 – 11

2.6.6 Status commands . . . . 2 – 14

2.7 Status model . . . . 2 – 17

2.7.1 Status byte (SB) . . . . 2 – 18

2.7.2 Service request enable (SRE) . . . . 2 – 18

2.7.3 Standard event status (ESR) . . . . 2 – 18

2.7.4 Port status event (PSEV) . . . . 2 – 19

General Information

Safety and Preparation for Use

⚠ WARNING **Dangerous voltages, capable of causing injury or death, are present in this instrument. Do not remove the product covers or panels. Do not apply power or operate the product without all covers and panels in place.**

AC line voltage

The universal input power supply of the SX199 Optical Interface Controller accommodates any voltage in the range 90 VAC to 260 VAC, with a frequency in the range 47 Hz to 63 Hz.

Line cord

The SX199 Optical Interface Controller has a detachable, three-wire power cord for connection to the power source and to a protective ground. The chassis of the instrument is connected to the outlet ground to protect against electrical shock. Always use an outlet which has a properly connected protective ground.

Fuse

The SX199 has an internal fuse that is not intended for service by the user.

If the *POWER* indicator does not illuminate when line power is applied and the power switch is in the "on" position, contact Stanford Research Systems for service.

Service

The SX199 Optical Interface Controller does not have any user serviceable parts inside. Refer service to a qualified technician.

Do not install substitute parts or perform any unauthorized modifications to this instrument. Contact the factory for instructions on how to return the instrument for authorized service and adjustment.

Symbols you may Find on SRS Products

Symbol	Description
⏦	Alternating current
⚡	Caution - risk of electric shock
⊥	Frame or chassis terminal
⚠	Caution - refer to accompanying documents
⏚	Earth (ground) terminal
⊣⊢	Battery
⏛	Fuse
⏽	On (supply)
○	Off (supply)

Notation

The following notation will be used throughout this manual.

⚠ WARNING

A warning means that injury or death is possible if the instructions are not obeyed.

⚠ CAUTION

A caution means that damage to the instrument or other equipment is possible.

Typesetting conventions used in this manual are:

- Front-panel indicators are set as *Overload*
- Remote command names are set as *IDN?
- Literal text other than command names is set as `OFF`

Remote command examples will all be set in monospaced font. In these examples, data sent by the host computer to the SX199 are set as `straight teletype font`, while responses received by the host computer from the SX199 are set as *`slanted teletype font`*.

Specifications

Rear panel

Parameter	Specification
Optical ports	Four (4), Avago “Versatile Link”, duplex, non-latching (mating part HFBR-4506Z)
Remote Interfaces	
RS-232	DB-9, 9600 / 57.6k baud, switch selected
GPIB	
Ethernet	10/100Base-T (auto)
Debug port	RS-232, DB-9

Optical ports

Parameter	Specification
Wavelength	660 nm (typ)
Fiber optic length	3 m standard (HFBR-RMD003Z) 10 m available (HFBR-RMD010Z)

General

Parameter	Specification
Temperature	0 °C to 40 °C, non-condensing
Power	25 W, 90 VAC to 260 VAC, 47 Hz to 63 Hz
Dimensions	8.25′′ W × 4′′ H × 11′′ D
Weight	5 lbs

1 Introduction

This chapter provides a basic overview of the SX199 Optical Interface Controller.

In This Chapter

1.1 Introduction to the instrument . . . . 1 – 2
1.1.1 Front Panel . . . . 1 – 2
1.1.2 Rear Panel . . . . 1 – 3
1.2 Remote configuration . . . . 1 – 3
1.2.1 IEEE-488 GPIB . . . . 1 – 3
1.2.2 Ethernet . . . . 1 – 5
1.2.3 RS-232 . . . . 1 – 6
1.3 Debug port . . . . 1 – 6

1.1 Introduction to the instrument

The SX199 Optical Interface Controller is a communications bridge to connect up to four (4) SRS instruments having a serial optical interface to a computer's remote interface via GPIB, ethernet, or RS-232.

1.1.1 Front Panel

The front panel of the SX199 (Figure 1.1) provides simple monitoring of the status of the remote interface and optical port connections. Whenever power is applied and the AC switch in the "ON" position, the *ON* power indicator should be illuminated.

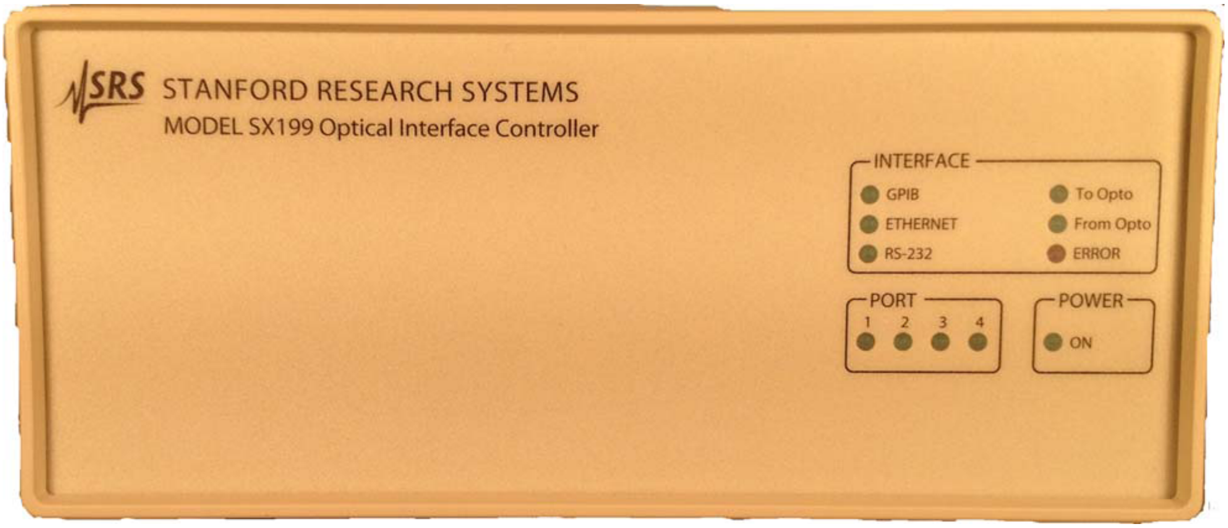
SRS STANFORD RESEARCH SYSTEMS
MODEL SX199 Optical Interface Controller
INTERFACE
GPIB
ETHERNET
RS-232
To Opto
From Opto
ERROR
PORT
1
2
3
4
POWER
ON

Figure 1.1: The SX199 front panel.

Indicators in the left-most column of the "INTERFACE" block indicate real-time data traffic on the three remote interfaces (GPIB, ethernet, and RS-232). Any data coming from or going to an interface will cause that interface indicator to flash. If data is being transmitted to one of the optical ports, then the *To Opto* indicator will also flash. If data is being received from one of the optical ports, the *From Opto* will illuminate. Command errors or buffer overruns will cause the *ERROR* indicator to illuminate. The *ERROR* indicator is cleared by the remote *ESR? query.

When the SX199 is in the "link" state with an optical port, the selected port is indicated by the "PORT" block on the front panel. The corresponding linked remote interface indicator is also illuminated.

1.1.2 Rear Panel

The rear panel of the SX199 (Figure 1.2) provides all external interface connections and the power switch. The unit is turned on when the AC power switch is in the "1" position, and off when the switch is in the "0" position.

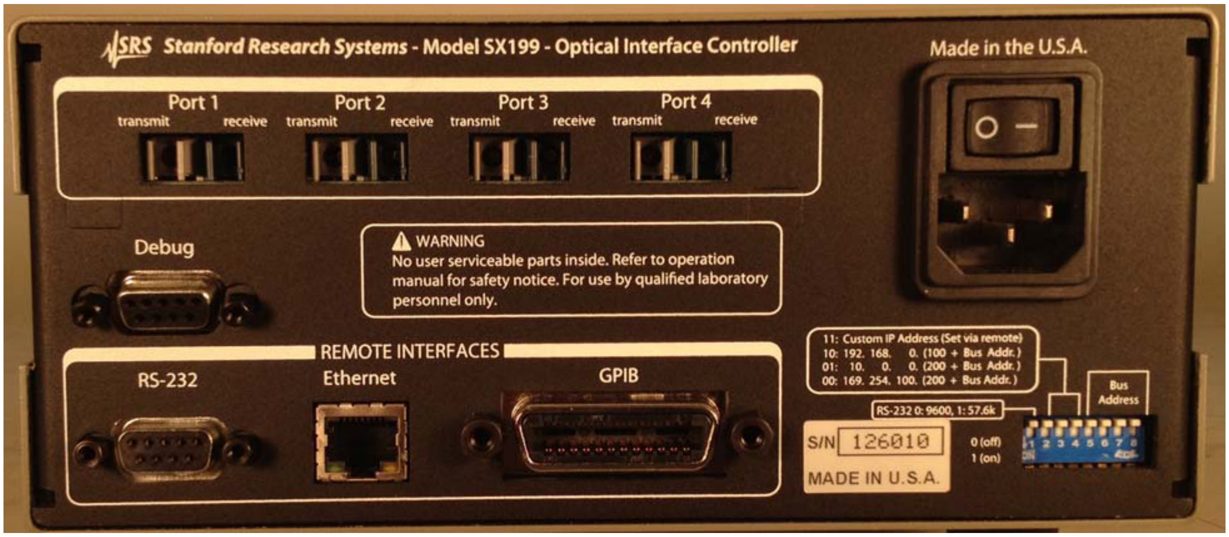
SRS Stanford Research Systems - Model SX199 - Optical Interface Controller
Made in the U.S.A.
Port 1
transmit
receive
Port 2
transmit
receive
Port 3
transmit
receive
Port 4
transmit
receive
Debug
WARNING
No user serviceable parts inside. Refer to operation manual for safety notice. For use by qualified laboratory personnel only.
REMOTE INTERFACES
RS-232
Ethernet
GPIB
11: Custom IP Address (Set via remote)
10: 192. 168. 0. (100 + Bus Addr.)
01: 10. 0. 0. (200 + Bus Addr.)
00: 169. 254. 100. (200 + Bus Addr.)
Bus Address
RS-232 0: 9600, 1: 57.6k
S/N 126010
MADE IN U.S.A.
0 (off)
1 (on)
1 2 3 4 5 6 7 8

Figure 1.2: The SX199 rear panel.

The optical ports are in the upper block of the rear panel, labeled "Port 1" through "Port 4". When mating the fiber optic cable to one of the port connections, be sure the small flange protrusion is oriented upwards (Figure 1.3. Once oriented and aligned, press the cable connector inwards until it "clicks" in place (Figure 1.4).

1.2 Remote configuration

Remote connection to the SX199 is supported with GPIB, RS-232 and Ethernet computer interfaces. These interfaces are configured by the rear-panel DIP switches.

1.2.1 IEEE-488 GPIB

The SX199 comes with an GPIB (IEEE-488) port for communicating over GPIB. The GPIB address is configured with the rightmost five positions of the larger DIP switch block; these position are labled "Bus Address."

The DIP switches encode the GPIB bus address as a binary number in the range 0 to 31. The labels printed around the DIP switch indicate

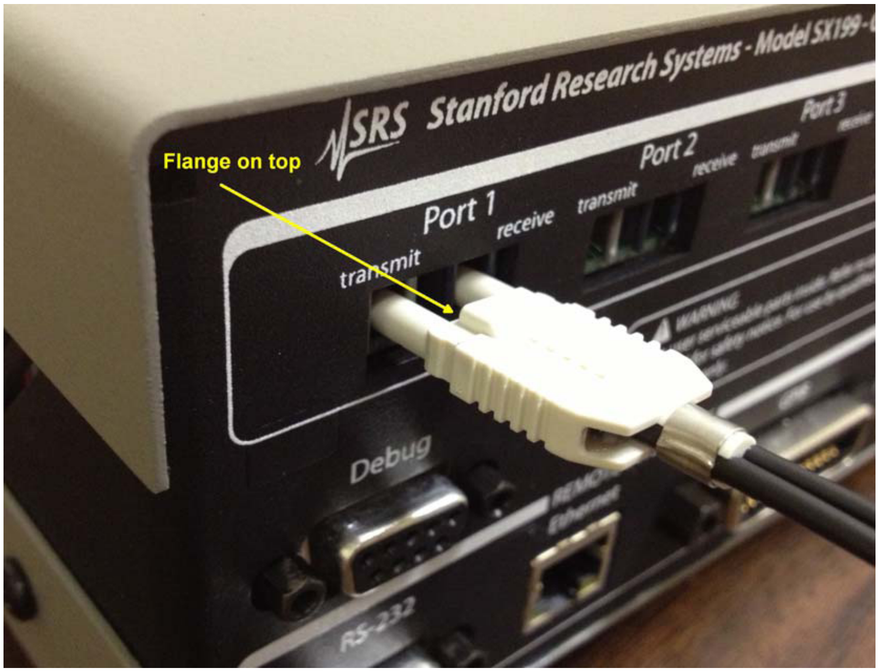

Figure 1.3: Orientation of optical fiber cable.

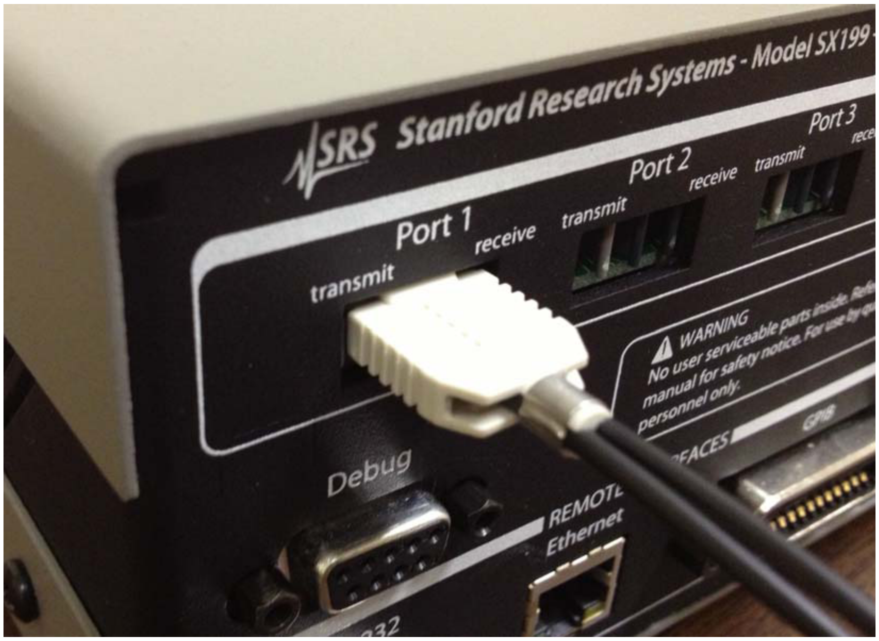

Figure 1.4: Fully seated optical fiber cable.

the polarity: a switch represents a binary '0' in the "up" position or a binary '1' in the "down" position, and the least-significant bit is on the right hand side. A setting of *up-up-down-up-down*, reading left to right on the bus address switches (as shown in Figure 1.2), represents a binary 00101, correspoding to bus address 5 (Figure 1.5).

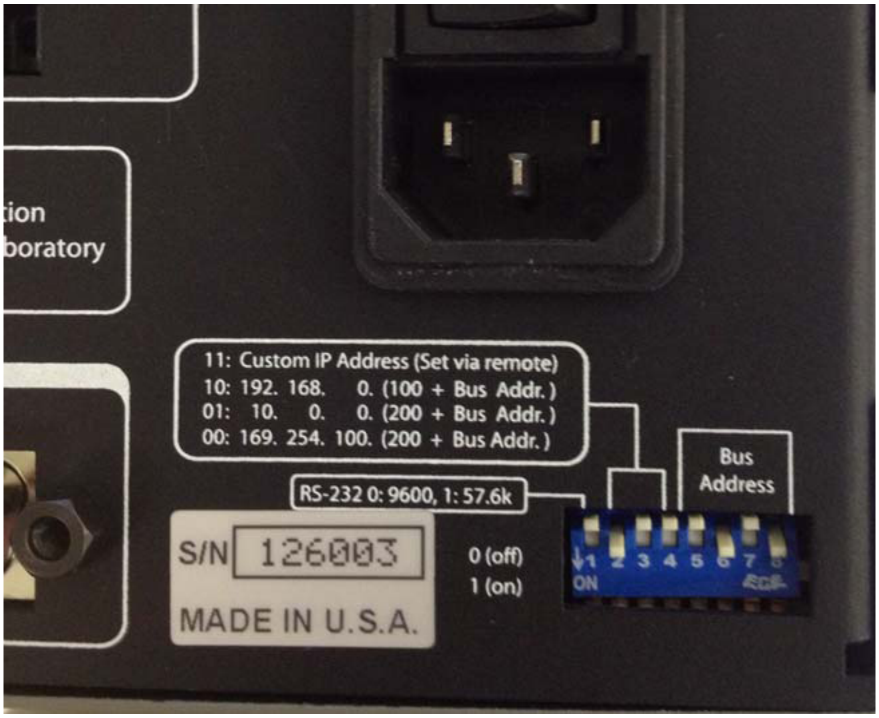
tion
boratory
11: Custom IP Address (Set via remote)
10: 192. 168. 0. (100 + Bus Addr.)
01: 10. 0. 0. (200 + Bus Addr.)
00: 169. 254. 100. (200 + Bus Addr.)
RS-232 0: 9600, 1: 57.6k
Bus Address
S/N 126003
MADE IN U.S.A.
0 (off)
1 (on)
1 2 3 4 5 6 7 8
ON

Figure 1.5: Rear panel DIP switch example.

The bus address should be set to a value between 1 and 30. If all five switches are in the 1 position, the value is forced to 30, not 31.

Any changes made will not take effect until the instrument is power cycled.

1.2.2 Ethernet

The SX199 comes standard with an RJ-45 network communications port located on the rear panel. The port may be used to communicate with the SX199 over a 10/100 Base-T ethernet connected network or LAN. The SX199 supports static network configuration as determined by the rear panel DIP switches. Any changes made to the interface configuration will not take effect until the unit is power cycled.

The IP Address configuration is primarily selected by positions 2 and 3 of the larger DIP switch. As with the GPIB address selection, a switch either represents a binary ‘0’ in the up position, or a binary ‘1’ in the down position. There are four possible ways to arrange these two switches: `00`, `01`, `10`, and `11`. The corresponding configurations are shown in Table 1.1.

Config. Switches	IP Address	Subnet Mask	Gateway
11	*Custom Address*	*Custom*	*Custom*
10	192.168.0.(100 + *Bus Address*)	255.255.255.0	0.0.0.0
01	172.201.25.(100 + *Bus Address*)	255.255.0.0	0.0.0.0
00	10.0.0.(200 + *Bus Address*)	255.0.0.0	0.0.0.0

Table 1.1: IP Address configuration

These switches give a number of possible static IP addresses that may be useful for your SX199, depending on your network environment. As an example, with the two switches set to 10 (*down-up*) and the (GPIB) Bus Address set to 5, the IP address will be configured as 192.168.0.105, the subnet mask will be 255.255.255.0, and the default gateway will be 0.0.0.0. Using a remote interface, it is also possible to assign and save custom values for these three parameters. Set the configuration switches to 11 to use your custom value.

1.2.3 RS-232

The RS-232 interface connector is a standard 9 pin, type D, female connector configured as a DCE (transmit on pin 3, receive on pin 2). The communication parameters are: 8 Data bits, 1 Stop bit, No Parity, No Hardware Flow Control. All of these parameters are fixed. The baud rate may be set to either 9600 or 57600 via the leftmost switch on the DIP switch block. Any changes made to the interface configuration will not take effect until the unit is power cycled.

1.3 Debug port

In addition to the optical ports and the remote interface connections, the SX199 includes an RS-232 “debugging” port on the rear panel. This connection, like the remote RS-232, is a standard 9 pin, type D, female connector configured as a DCE. The baud rate for the debug port matches the main RS-232 port setting.

All data passing through the SX199, whether part of link mode to or from an optical port, or native SX199 commands and queries, is also transmitted out the debug port. This debug stream can be convenient when troubleshooting new user programs or instrument configurations.

Error messages from the SX199 are also transmitted on the debug port.

2 Remote Operation

This chapter describes operating the SX199 over the remote interfaces.

In This Chapter

2.1 Index of commands . . . . 2 – 2
2.2 Alphabetic list of commands . . . . 2 – 3
2.3 Introduction . . . . 2 – 5
2.3.1 Interface configuration . . . . 2 – 5
2.3.2 Buffers . . . . 2 – 5
2.4 Ethernet . . . . 2 – 5
2.5 Link model . . . . 2 – 6
2.6 Commands . . . . 2 – 7
2.6.1 Command syntax . . . . 2 – 7
2.6.2 Notation . . . . 2 – 8
2.6.3 Examples . . . . 2 – 8
2.6.4 Link commands . . . . 2 – 9
2.6.5 Interface commands . . . . 2 – 11
2.6.6 Status commands . . . . 2 – 14
2.7 Status model . . . . 2 – 17
2.7.1 Status byte (SB) . . . . 2 – 18
2.7.2 Service request enable (SRE) . . . . 2 – 18
2.7.3 Standard event status (ESR) . . . . 2 – 18
2.7.4 Port status event (PSEV) . . . . 2 – 19

2.1 Index of commands

Symbol	Definition
i, j	Unsigned integer
z	Literal token
(?)	Required for queries; illegal for set commands
var	Parameter always required
{*var*}	Required parameter for set commands; illegal for queries
[*var*]	Optional parameter for both set and query forms

Link

LINK(?) {*i*}	2 – 9	Link
LNKS(?) {*i*}	2 – 9	Link Serial
LNKE(?) {*i*}	2 – 10	Link Ethernet
LNKG(?) {*i*}	2 – 10	Link GPIB
UNLK	2 – 10	Unlink
SESC(?) {*i*}	2 – 10	Set escape character

Interface

*IDN?	2 – 11	Identify
TOKN(?) {*z*}	2 – 11	Token Mode
TERM(?) {*z*}	2 – 11	Response Termination
*OPC(?)	2 – 11	Operation complete
ULOC(?) {*i*}	2 – 12	Unlock Ethernet
IPAD(?) i {, *j*}	2 – 12	IP Address
NMSK(?) i {, *j*}	2 – 12	Netmask
GWAY(?) i {, *j*}	2 – 13	Default Gateway
MACA?	2 – 13	Ethernet hardware address
ENET(?) {*z*}	2 – 13	Ethernet speed
SPAR {*z*}	2 – 13	Save User Parameters
*RST	2 – 13	Reset

Status

*STB? [*i*]	2 – 14	Status byte
*SRE(?) [*i*,] {*j*}	2 – 14	Service request enable
*ESR? [*i*]	2 – 14	Standard event status
*ESE(?) [*i*,] {*j*}	2 – 14	Standard event status enable
PSEV? [*i*]	2 – 14	Port status event
PSEN(?) [*i*,] {*j*}	2 – 14	Port status enable
*CLS	2 – 15	Clear status
LEXE?	2 – 15	Last execution error
LCME?	2 – 15	Last command error

2.2 Alphabetic list of commands

★		
*CLS	2 – 15	Clear status
*ESE(?) [i,] {j}	2 – 14	Standard event status enable
*ESR? [i]	2 – 14	Standard event status
*IDN?	2 – 11	Identify
*OPC(?)	2 – 11	Operation complete
*RST	2 – 13	Reset
*SRE(?) [i,] {j}	2 – 14	Service request enable
*STB? [i]	2 – 14	Status byte
E		
ENET(?) {z}	2 – 13	Ethernet speed
G		
GWAY(?) i {, j}	2 – 13	Default Gateway
I		
IPAD(?) i {, j}	2 – 12	IP Address
L		
LCME?	2 – 15	Last command error
LEXE?	2 – 15	Last execution error
LINK(?) {i}	2 – 9	Link
LNKE(?) {i}	2 – 10	Link Ethernet
LNKG(?) {i}	2 – 10	Link GPIB
LNKS(?) {i}	2 – 9	Link Serial
M		
MACA?	2 – 13	Ethernet hardware address
N		
NMSK(?) i {, j}	2 – 12	Netmask
P		
PSEN(?) [i,] {j}	2 – 14	Port status enable
PSEV? [i]	2 – 14	Port status event
S		
SESC(?) {i}	2 – 10	Set escape character
SPAR {z}	2 – 13	Save User Parameters

T

TERM(?) {*z*}	2 – 11	Response Termination
TOKN(?) {*z*}	2 – 11	Token Mode

U

ULOC(?) {*i*}	2 – 12	Unlock Ethernet
UNLK	2 – 10	Unlink

2.3 Introduction

The SX199 Optical Interface Controller provides buffered, multiplexed communications between the host computer and up to 4 instruments through an optical fiber interface. Data is encoded on the optical fiber as asynchronous serial data using standard UART timing protocol, with 8 data bits, no parity, one stop bit, and 9600 baud. Data polarity on the optical ports is: illuminated = start bit = data "0".

The host computer communicates with the SX199 through the host "remote interface", which can be either GPIB, ethernet, or RS-232. All remote interfaces are active and available simultaneously on the SX199.

Remote operation of the SX199 is through a simple command language documented in this chapter. Both set and query forms of most commands are supported, allowing the user complete control of the controller from a remote computer through the GPIB, RS-232, or ethernet interface. While the SX199 has no direct instrumentation itself, it acts as a "transparent link" to one of up to four downstream instruments connected through the optical fiber interface.

2.3.1 Interface configuration

All three remote interfaces have configuration settings that may need adjusting from the rear panel DIP switch. See section 1.2 on page 1 – 3 for configuration of the specific switch settings.

2.3.2 Buffers

Except during link mode, the SX199 stores incoming bytes from the remote interfaces in separate 64-byte input buffers. Characters accumulate in the input buffer until a command terminator (⟨CR⟩, ⟨LF⟩, or GPIB-EOI signal) is received, at which point the message is parsed and enqueued for execution. Query responses from the SX199 are buffered in interface-specific 256-byte output queues.

If an input buffer overflows, then all data in the input buffer are discarded, and an error is recorded in the ESR status register.

2.4 Ethernet

To connect to the SX199 via the ethernet interface, the user must first configure the network parameters. Once connected to local network (or simple ethernet crossover cable), establish a socket connection to TCP/IP port 8888. Send a blank line to initialize the connection, and then send the ULOC 1 command to unlock the ethernet command processor (see page 2 – 12). Next, the *IDN? query should be sent. The SX199 should reply with the response string described on page 2 – 11 .

2.5 Link model

The SX199 uses a "link" framework for providing communications between a host remote interface and the downstream instruments connected by optical fiber. In this link model, when a "link" is established, a *single* remote interface is linked to a *single* optical port: data bytes received from the remote interface are relayed directly to the optical fiber port, and response data received from the optical port are relayed back to the remote interface. While linked, front panel indicators for the selected optical port and the linked remote interface are illuminated.

After first establishing a link, it may be necessary to initialize the optical port and the remote interface of the fiber-coupled instrument by sending a ⟨LF⟩ character before beginning remote commanding.

The host remote interface(s) that are *not* linked remain available for regular commanding to the SX199. These interface(s) can be used to reconfigure the link state, or to query status registers of the SX199, or any other remote command documented in this chapter. The linked remote interface, however, will not be processed (parsed) by the SX199—commands transmitted to the SX199 via the linked remote interface will be relayed byte-for-byte to the linked optical port, and *not* interpreted as SX199 commands.

The link state can be exited by transmitting an "escape" character from the host computer to the linked remote interface, followed by any other character. To transmit the escape character to the linked optical port, you must transmit the escape character twice. After reset, the escape character is the "!" character (character code 33). See the SESC command (page 2 – 10) for reprogramming the escape character to another byte.

2.6 Commands

This section provides syntax and operational descriptions for remote commands.

2.6.1 Command syntax

The four letter mnemonic (shown in CAPS) in each command sequence specifies the command. The rest of the sequence consists of parameters.

Commands may take either *set* or *query* form, depending on whether the "?" character follows the mnemonic. *Set only* commands are listed without the "?", *query only* commands show the "?" after the mnemonic, and *optionally query* commands are marked with a "(?)".

Parameters shown in { } and [ ] are not always required. Parameters in { } are required to set a value, and should be omitted for queries. Parameters in [ ] are optional in both set and query commands. Parameters listed without surrounding characters are always required.

Do *not* send ( ) or { } or [ ] as part of the command.

Multiple parameters are separated by commas. Multiple commands may be sent on one command line by separating them with semicolons (;) so long as the input buffer does not overflow. Commands are terminated by either ⟨CR⟩ or ⟨LF⟩ characters. Null commands and whitespaces are ignored. Execution of the command does not begin until the command terminator is received.

tokens *Token* parameters (generically shown as *z* in the command descriptions) can be specified either as a keyword or as an integer value. Command descriptions list the valid keyword options, with each keyword followed by its corresponding integer value. For example, to set the response terminator to ⟨LF⟩, the following two commands are equivalent:

TERM LF —or— TERM 2

For queries that return token values, the return format (keyword or integer) is specified with the TOKN command.

2.6.2 Notation

The following table summarizes the notation used in the command descriptions:

Symbol	Definition
i, j	Unsigned integer
z	Literal token
(?)	Required for queries; illegal for set commands
var	Parameter always required
{*var*}	Required parameter for set commands; illegal for queries
[*var*]	Optional parameter for both set and query forms

2.6.3 Examples

Each command is provided with a simple example illustrating its usage. In these examples, all data sent by the host computer to the SX199 are set as `straight teletype font`, while responses received by the host computer from the SX199 are set as *`slanted teletype font`*.

The usage examples vary with respect to set/query, optional parameters, and token formats. These examples are not exhaustive, and are intended to provide a convenient starting point for user programming.

2.6.4 Link commands

LINK(?) {*i*}

Link

Set or query the optical port link.

Setting the LINK *i* command to a value *i* between 1 and 4 establishes the link state between optical port *i* and the remote interface on which the LINK command was received by the SX199. Once executed, that remote interface will now be "linked" — subsequent data received from the host computer on the remote interface will be transmitted to optical port *i*.

To terminate a link session *from* the linked remote interface, send the escape character followed by any other charater. After reset, the escape character is initially the "!" character (character code 33).

The query form LINK? responds with 0 if no link is presently active, or with a two-digit integer *rp*, where the first digit *r* is the linked remote interface (1 for RS-232, 2 for GPIB, and 3 for ethernet). The second digit *p* is the linked optical port, and is a value from 1 to 4.

The following query example shows the response when the GPIB interface is linked to optical port 4.

Example:

```
LINK?
24
```

LNKS(?) {*i*}

Link Serial

Set (query) the optical port link status for the host RS-232 (serial) port {to optical port *i*}.

Setting LNKS to a value *i* between 1 and 4 establishes an active link between the RS-232 (serial) remote interface and optical port *i*. If any other port was previously linked, it disconnects that link in the process. If LNKS is received on the RS-232 remote interface, this is equivalent to the LINK command.

Setting LNKS 0 disconnects the link state of the RS-232 (serial) remote interface from any optical port. Note if either of the other two remote interfaces is in the active link state, LNKS 0 has no effect.

Example:

```
LNKS 1
```

LNKE(?) {*i*}

Link Ethernet

Set (query) the optical port link status for the host ethernet port {to optical port *i*}.

Setting LNKE to a value *i* between 1 and 4 establishes an active link between the ethernet remote interface and optical port *i*. If any other port was previously linked, it disconnects that link in the process. If LNKE is received on the ethernet remote interface, this is equivalent to the LINK command.

Setting LNKE `0` disconnects the link state of the ethernet remote interface from any optical port. Note if either of the other two remote interfaces is in the active link state, LNKE `0` has no effect.

Example:

```
LNKE 1
```

LNKG(?) {*i*}

Link GPIB

Set (query) the optical port link status for the host GPIB port {to optical port *i*}.

Setting LNKG to a value *i* between 1 and 4 establishes an active link between the GPIB remote interface and optical port *i*. If any other port was previously linked, it disconnects that link in the process. If LNKG is received on the GPIB remote interface, this is equivalent to the LINK command.

Setting LNKG `0` disconnects the link state of the ethernet remote interface from any optical port. Note if either of the other two remote interfaces is in the active link state, LNKG `0` has no effect.

Example:

```
LNKG 1
```

UNLK

Unlink

This set-only command forces the SX199 to the unlinked state.

Note that this command can only be received on a currently-unlinked remote interface. To unlink using the linked remote interface, transmit the escape character `!` followed by any other character.

Example:

```
UNLK
```

SESC(?) {*i*}

Set escape character

Set (query) the escape character {to character code *i*}. The valid range for *i* is $0 \leq i < 255$. The default value is SESC `33` (the “`!`” character).

Example:

```
SESC?
33
```

2.6.5 Interface commands

*IDN?

Identify

Query the SX199 identification string.

The response is formatted as:
`Stanford␣Research␣Systems,SX199,s/n******,ver#.##`
where `******` is the 6-digit serial number, and `#.##` is the firmware revision level.

Example:

```
*IDN?
Stanford␣Research␣Systems,SX199,s/n098023,ver1.01
```

TOKN(?) {*z*}

Token Mode

Set (query) the token response mode {to *z*=(**`OFF 0`**, `ON 1`)}.

Token response mode controls the formatting of response messages generated by the SX199 to remote queries of token-type values. When TOKN `OFF`, the SX199 responds with the numeric version of the token quantity. When TOKN `ON`, the text version is returned.

Example:

```
TOKN?
ON
```

TERM(?) {*z*}

Response Termination

Set (query) the token response mode {to *z*=(`NONE 0`, `CR 1`, `LF 2`, `CRLF 3`, `LFCR 4`)}.

Response messages generated by the SX199 will be terminated by the 0-, 1- or 2-character termination sequence specified by TERM. Note that the TERM command is *interface specific*. In other words, if TERM `LF` is received on the RS-232 interface, and then TERM `CRLF` is received on the ethernet interface, then queries received on the RS-232 interface shall generate response messages terminated with the LF character, while queries received on the ethernet interface shall generate response messages terminated by the CR and LF characters.

Example:

```
TERM LF
```

*OPC(?)

Operation complete

The set form, *OPC, will set the OPC bit in the Standard Event Status register; the query form, *OPC?, will return the value 1.

*OPC is useful for pacing streams of remote commands; the *OPC command will not be processed by the command execution of the SX199 until all preceding commands have been executed.

Example:
```
*OPC?
1
```

ULOC(?) {*i*}

Unlock Ethernet

Set (query) the ethernet command lockout {to *i*}.

Upon power-up, the SX199 defaults to ULOC 0, which *locks out* all remote commanding over the ethernet port. To enable ethernet control, send the command ULOC 1.

When first connecting to the ethernet command port (port 8888), the user's application program should begin by sending ULOC 1. The ULOC command is the only command that can be processed over ethernet while ULOC 0.

Example:
```
ULOC 1
```

IPAD(?) i {, *j*}

IP Address

Set (query) byte *i* of the "Custom" internet address {to *j*}.

Note that byte 0 is the left-most byte of the address. This address is used when the rear-panel DIP switches are in the "11" (*down-down*) position.

Also note that changes to IPAD are *not* saved until the SPAR 0 command is issued.

Example:
```
IPAD?0; IPAD?1; IPAD? 2; IPAD?3
169;254;46;27
```

NMSK(?) i {, *j*}

Netmask

Set (query) byte *i* of the "Custom" internet network mask {to *j*}.

Note that byte 0 is the left-most byte of the mask. This address is used when the rear-panel DIP switches are in the "11" (*down-down*) position.

Also note that changes to NMSK are *not* saved until the SPAR 0 command is issued.

Example:
```
NMSK 1, 255; NMSK?0; NMSK?1; NMSK?2; NMSK3
255;255;0;0
```

GWAY(?) i {, *j*}

Default Gateway

Set (query) byte *i* of the internet default gateway {to *j*}.

Note that byte `0` is the left-most byte of the address. This address is used when the rear-panel DIP switches are in the "11" (*down-down*) position.

Also note that changes to GWAY are *not* saved until the SPAR `0` command is issued.

Example:

```
GWAY 0, 172
```

MACA?

Ethernet hardware address

Query the low-level ethernet hardware address (MAC address). This is not the same as the IP address, and cannot be changed by the user.

Example:

```
MACA?
0019:b303:ffff
```

ENET(?) {*z*}

Ethernet speed

Set (query) the ethernet speed{to *z*=(**AUTO 0**, `M10 1`, `M100 2`)}.

The ethernet speed can be set to 100Base-T (*z* = `M100`), 10Base-T (*z* = `M10`), or autonegotiate between the two speeds (*z* = `AUTO`).

Example:

```
ENET?
0
```

SPAR {*z*}

Save User Parameters

Save user settings to non-volatile memory. The token

z should always be `0`. This command is needed to record changes to internet address before power cycling.

Example:

```
SPAR 0
```

*RST

Reset

Reset the SX199 to its default configuration.

The following commands are internally excecuted upon receipt of the *RST command:

- UNLK
- TOKN OFF

Note that *RST does not modify the SESC setting, or any of the status enable register values.

Example: `*RST`

2.6.6 Status commands

---

*STB? [*i*]

Status byte

Reads the Status Byte register [bit *i*].

Example:

```
*STB?
0
```

---

*SRE(?) [*i*,] {*j*}

Service request enable

Set (query) the Service Request Enable register [bit *i*] {to *j*}.

Example: `*SRE 0,1`

---

*ESR? [*i*]

Standard event status

Reads the Standard Event Status Register [bit *i*].

Upon executing *ESR?, the returned bit(s) of the ESR register are cleared.

Example:

```
*ESR?
64
```

---

*ESE(?) [*i*,] {*j*}

Standard event status enable

Set (query) the Standard Event Status Enable Register [bit *i*] {to *j*}.

Example:

```
*ESE 6,1
ESE?
64
```

PSEV? [*i*]

Port status event

Reads the Port Status Event Register [bit *i*].

Upon executing PSEV?, the returned bit(s) of the PSEV register are cleared.

Example:

```
PSEV?
4
```

PSEN(?) [*i*,] {*j*}

Port status enable

Set (query) the Port Status Enable Register [bit *i*] {to *j*}.

Example:

```
PSEN 3,1
PSEN?
4
```

*CLS

Clear status

*CLS immediately clears the ESR register and the PSEV register.

Example:

```
*CLS
```

LEXE?

Last execution error

Query the last execution error code. A query of LEXE? always clears the error code, so a subsequent LEXE? will return `0`. Valid codes are:

Value	Definition
0	No execution error since last LEXE?
1	Illegal value
2	Wrong token
3	Invalid bit
4	Queue full
5	Not compatible

Example:

```
LNKG 7; LEXE?; LEXE?
1;0
```

The error (1, "Illegal value,") is because the parameter value (7) is too large for LNKG. The second read of LEXE? returns `0`.

LCME?

Last command error

Query the last command error code. A query of LCME? always clears the error code, so a subsequent LCME? will return `0`. Valid codes are:

Value	Definition
0	No execution error since last LCME?
1	Illegal command
2	Undefined command
3	Illegal query
4	Illegal set
5	Missing parameter(s)
6	Extra parameter(s)
7	Null parameter(s)
8	Parameter buffer overflow
9	Bad floating-point
10	Bad integer
11	Bad integer token
12	Bad token value
13	Bad hex block
14	Unknown token

Example:

```
*IDN
LCME?
```

4

The error (4, “Illegal set”) is due to the missing “?”.

2.7 Status model

status registers The SX199 status registers follow the hierarchical IEEE–488.2 format. A block diagram of the status register array is given in Figure 2.1.

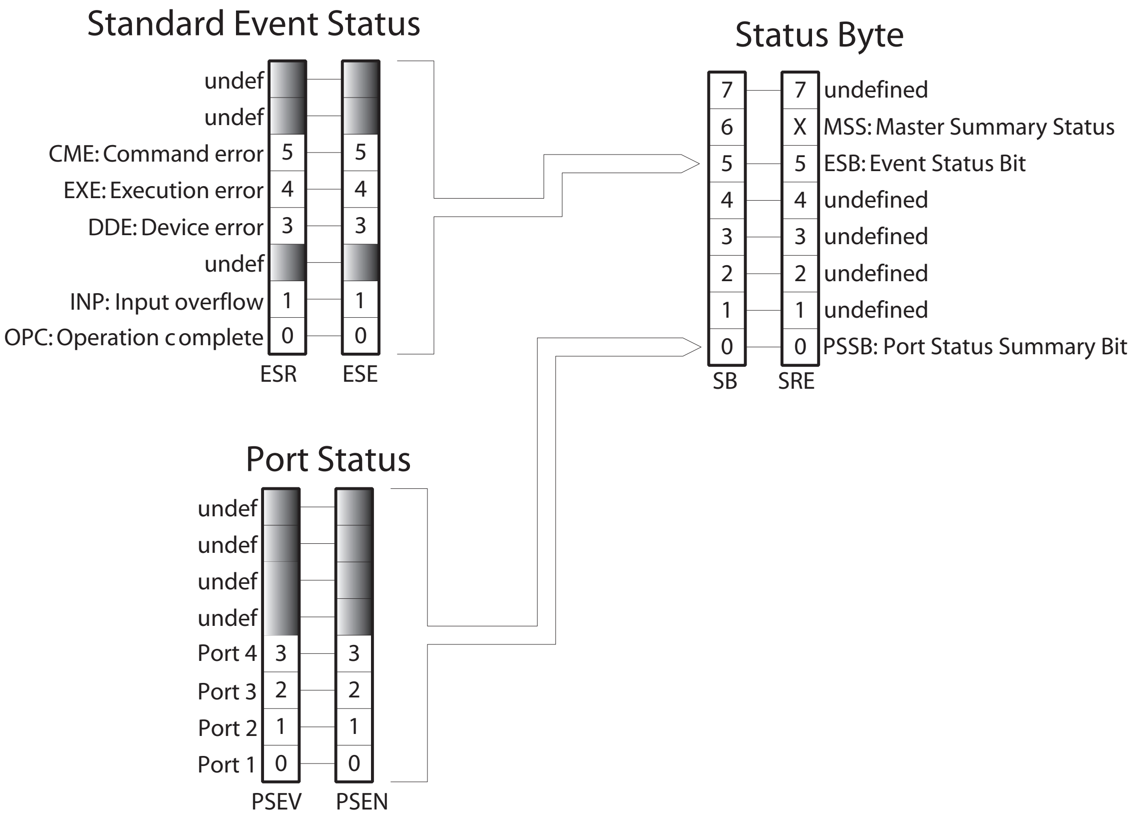

Figure 2.1: Status Model for the SX199 Optical Interface Controller

There are three categories of registers in the status model of the controller:

Event Registers : These read-only registers record the occurrence of defined events within the controller. If the event occurs, the corresponding bit is set to 1. Upon querying an event register, any set bits within it are cleared. These are sometimes known as “sticky bits,” since once set, a bit can only be cleared by reading its value. Event register names end with SR or EV.

Enable Registers : These read/write registers define a bitwise mask for their corresponding event register. If any bit position is set in an event register while the same bit position is also set in the enable register, then the corresponding summary bit message is set in the Status Byte. Enable register names end with SE or EN.

Status Byte : This read-only register represents the top of the status model, and is populated with summary bit messages and interface condition bits. Enabled bits within the Status Byte generate the remote Request Service event.

At power-on, all status registers are cleared.

2.7.1 Status byte (SB)

The Status Byte is the top-level summary of the SX199 status model. When enabled by the Service Request Enable register, a bit set in the Status Byte causes the MSS (Master Summary Status) bit to be set.

Weight	Bit	Flag
1	0	PSSB
2	1	undef (0)
4	2	undef (0)
8	3	undef (0)
16	4	undef (0)
32	5	ESB
64	6	MSS
128	7	undef (0)

PSSB : Port Status Summary Bit. Indicates whether one or more of the enabled event bits in the Port Status Event Register is true.

ESB : Event Status Bit. Indicates whether one or more of the enabled events in the Standard Event Status Register is true.

MSS : Master Summary Status. Indicates whether one or more of the enabled status messages in the Status Byte register is true.

This register is read with the *STB? query.

2.7.2 Service request enable (SRE)

Each bit in the SRE corresponds one-to-one with a bit in the SB register, and acts as a bitwise AND of the SB flags to generate MSS. Bit 6 of the SRE is undefined—setting it has no effect, and reading it always returns 0. This register is set and queried with the *SRE(?) command.

At power-on, this register is cleared.

2.7.3 Standard event status (ESR)

The Standard Event Status Register consists of 8 event flags. These event flags are all “sticky bits” that are set by the corresponding events, and cleared only by reading or with the *CLS command. Reading a single bit (with the *ESR? *i* query) clears only Bit *i*.

Weight	Bit	Flag
1	0	OPC
2	1	INP
4	2	undef (0)
8	3	DDE
16	4	EXE
32	5	CME
64	6	undef (0)
128	7	undef (0)

OPC : Operation Complete. Set by the *OPC command.

INP : Input Overflow. Indicates data in one of the remote interface input buffers has been lost due to buffer overflow.

DDE : Device-Dependent Error. Indicates an internal command queue overflow.

EXE : Execution Error. Indicates the error in a command that was successfully parsed. Out-of-range parameters are an example.

CME : Command Error. Indicates a command parser-detected error.

2.7.3.1 Standard event status enable (ESE)

The ESE acts as a bitwise AND with the ESR register to produce the single-bit ESB message in the Status Byte Register (SB). The register can be set and queried with the *ESE(?) command.

At power-on, this register is cleared.

2.7.4 Port status event (PSEV)

The Port Status Event Register consists of 4 event flags. These event flags are all "sticky bits" that are set by the corresponding events, and cleared only by reading or with the *CLS command. Reading a single bit (with the PSEV? *i* query) clears only Bit *i*.

Weight	Bit	Flag
1	0	Port 1
2	1	Port 2
4	2	Port 3
8	3	Port 4
16	4	undef (0)
32	5	undef (0)
64	6	undef (0)
128	7	undef (0)

The four defined bits in PSEV correspond one-for-one with the four optical ports. A bit is set in PSEV when the corresponding optical port asserts the "status message" by driving a serial "break" signal on the optical port. This is the method available for optical port remote instruments to request service asynchronously.

2.7.4.1 Port status event enable (PSEN)

The PSEN acts as a bitwise AND with the PSEV register to produce the single-bit PSSB message in the Status Byte Register (SB). The register can be set and queried with the PSEN(?) command.

At power-on, this register is cleared.