

# **SERIAL ATTACHED SCSI (SAS) CONSORTIUM**

**Clause 6  
SAS SPL Link Layer Test Suite  
Version 1.4**

*Technical Document*



*Last Updated: 1 February 2016*

---

*Serial Attached SCSI Consortium  
InterOperability Laboratory  
University of New Hampshire*

*121 Technology Drive, Suite 2  
Durham, NH 03824  
Phone: (603) 862-3582  
Fax: (603) 862-4181  
<http://www.ioi.unh.edu/consortiums/sas>*

---

**TABLE OF CONTENTS**

<b>TABLE OF CONTENTS</b> .....	<b>2</b>
<b>MODIFICATION RECORD</b> .....	<b>5</b>
<b>ACKNOWLEDGMENTS</b> .....	<b>7</b>
<b>INTRODUCTION</b> .....	<b>8</b>
<b>REFERENCES</b> .....	<b>10</b>
<b>GROUP 1: LINK LAYER TESTS FOR TARGETS</b> .....	<b>11</b>
TEST 6.1.1 - LINK RESET – REPEAT PHY SEQUENCE IF NO IDENTIFY RECEIVED ....	12
TEST 6.1.2 - LINK RESET – IGNORE ADDITIONAL IDENTIFY RECEIVED .....	13
TEST 6.1.3 - LINK RESET – HARD_RESET RECEIVED .....	14
TEST 6.1.4 - CONNECTIONS – OPEN_ACCEPT .....	15
TEST 6.1.5 - CONNECTIONS – 2 SOAF RECEIVED.....	16
TEST 6.1.6 - CONNECTIONS - OPEN_REJECT CONNECTION RATE NOT SUPPORTED .....	17
TEST 6.1.7 - CONNECTIONS - OPEN_REJECT PROTOCOL NOT SUPPORTED .....	18
TEST 6.1.8 - CONNECTIONS - OPEN_REJECT WRONG DESTINATION .....	19
TEST 6.1.9 - CONNECTIONS - OPEN_REJECT RETRY.....	20
TEST 6.1.10 - SSP_FRAMES – INTERLOCKED FRAME .....	21
TEST 6.1.11 - SSP_FRAMES - NO ACK FOR INTERLOCKED FRAME.....	22
TEST 6.1.12 - SSP_FRAMES – MULTIPLE ACKS .....	23
TEST 6.1.13 - SSP FRAMES - RRDY.....	24
TEST 6.1.14 - CLOSING SSP CONNECTIONS – DONE (NORMAL).....	25
TEST 6.1.15 - CLOSING SSP CONNECTIONS – DONE (ACK/NAK TIMEOUT) .....	26
TEST 6.1.16 - CLOSING SSP CONNECTIONS – DONE (CREDIT TIMEOUT).....	27
TEST 6.1.17 - CLOSING SSP CONNECTIONS – CLOSE (NORMAL) .....	28
TEST 6.1.18 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY TESTING STATION .....	29
TEST 6.1.19 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY DUT .....	30
TEST 6.1.20 - CONNECTIONS THROUGH EXPANDERS – OPEN SSP TARGET .....	31
TEST 6.1.21 - BREAK – OPENTIMEOUT TIMER SSP TARGET .....	32
TEST 6.1.22 - CLOSING SSP CONNECTIONS – CLOSE (CLEAR AFFILIATION) .....	33
TEST 6.1.23 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 0) .....	34
TEST 6.1.24 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 1) .....	35
TEST 6.1.25 - LINK RESET – HARD_RESET RECIVED AFTER IDENTIFY.....	36
TEST 6.1.26 - SSP_FRAMES – INTERLOCKED FRAME / NAK.....	37
<b>GROUP 2: LINK LAYER TESTS FOR INITIATORS</b> .....	<b>38</b>
TEST 6.2.1 - LINK RESET – REPEAT PHY SEQUENCE IF NO IDENTIFY RECEIVED ....	39
TEST 6.2.2 - LINK RESET – IGNORE ADDITIONAL IDENTIFY RECEIVED .....	40
TEST 6.2.3 - LINK RESET – HARD_RESET RECEIVED .....	41

*The University of New Hampshire  
InterOperability Laboratory*

TEST 6.2.4 - LINK RESET – SMP INITIATOR TO PERFORM DISCOVER .....	42
TEST 6.2.5 - LINK RESET – EDGE EXPANDER TRANSMITS OPEN_REJECT .....	43
TEST 6.2.6 - CONNECTIONS – OPEN_ACCEPT .....	44
TEST 6.2.7 - CONNECTIONS – 2 DIFFERENT SOAF RECEIVED .....	45
TEST 6.2.8 - CONNECTIONS - OPEN_REJECT CONNECTION RATE NOT SUPPORTED .....	46
TEST 6.2.9 - CONNECTIONS - OPEN_REJECT PROTOCOL NOT SUPPORTED .....	47
TEST 6.2.10 - CONNECTIONS - OPEN_REJECT WRONG DESTINATION .....	48
TEST 6.2.11 - CONNECTIONS - OPEN_REJECT RETRY .....	49
TEST 6.2.12 - SSP_FRAMES – MULTIPLE ACKs .....	50
TEST 6.2.13 - SSP FRAMES - RRDY .....	51
TEST 6.2.14 - CLOSING SSP CONNECTIONS – DONE (NORMAL) .....	52
TEST 6.2.15 - CLOSING SSP CONNECTIONS – DONE (ACK/NAK TIMEOUT) .....	53
TEST 6.2.16 - CLOSING SSP CONNECTIONS – DONE (CREDIT TIMEOUT) .....	54
TEST 6.2.17 - CLOSING SSP CONNECTIONS – CLOSE (NORMAL) .....	55
TEST 6.2.18 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY TESTING STATION .....	56
TEST 6.2.19 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY DUT .....	57
TEST 6.2.20 - CONNECTIONS THROUGH EXPANDER – OPEN SSP INITIATOR .....	58
TEST 6.2.21 - CONNECTIONS THROUGH EXPANDERS – OPEN SMP INITIATOR .....	59
TEST 6.2.22 - CLOSING SSP CONNECTIONS – CLOSE (CLEAR AFFILIATION) .....	60
TEST 6.2.23 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 0) .....	61
TEST 6.2.24 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 1) .....	62
<b>GROUP 3: LINK LAYER TESTS FOR EXPANDERS .....</b>	<b>63</b>
TEST 6.3.1 - LINK RESET – REPEAT PHY SEQUENCE IF NOT IDENTIFY RECEIVED ..	64
TEST 6.3.2 - IGNORE ADDITIONAL IDENTIFY RECEIVED .....	65
TEST 6.3.3 - LINK RESET – HARD_RESET RECEIVED .....	66
TEST 6.3.4 - CONNECTIONS – OPEN_ACCEPT .....	67
TEST 6.3.5 - CONNECTIONS – 2 SOAF RECEIVED .....	68
TEST 6.3.6 - CONNECTIONS - OPEN_REJECT NO DESTINATION .....	69
TEST 6.3.7 - CONNECTIONS - OPEN_REJECT BAD DESTINATION .....	70
TEST 6.3.8 - CONNECTIONS – OPEN STP TARGET .....	71
TEST 6.3.9 - CONNECTIONS – OPEN REJECT (STP RESOURCES BUSY) .....	72
TEST 6.3.10 - CONNECTIONS – SATA_HOLD .....	73
TEST 6.3.11 - CONNECTIONS – SATA_RIP .....	74
TEST 6.3.12 - BREAK – OPENTIMEOUT TIMER STP TARGET .....	75
TEST 6.3.13 - BREAK – CONNECTION REQUEST NOT TRANSMITTED .....	76
TEST 6.3.14 - BREAK – CONNECTION REQUEST TRANSMITTED .....	77
TEST 6.3.15 - AIP – TRANSMITTED .....	78
TEST 6.3.16 - AIP – OPEN RECEIVED HIGHER PRIORITY .....	79
TEST 6.3.17 - AIP – OPEN RECEIVED LOWER PRIORITY .....	80
TEST 6.3.18 - AWT TIMER – ARBITRATION WAIT TIME .....	81
TEST 6.3.19 - AWT TIMER – RESET TO ZERO .....	82

*The University of New Hampshire  
InterOperability Laboratory*

TEST 6.3.20 - AWT_TIMER – ARBITRATION FAIRNESS .....	83
TEST 6.3.21 - OPEN_REJECT – PATHWAY BLOCKED .....	84
TEST 6.3.22 - SMP_REQUEST – PROPER RESPONSE FORMAT .....	85
TEST 6.3.23 - SMP_REQUEST – CLOSE .....	86
TEST 6.3.25 - CLOSING SSP CONNECTIONS – CLOSE (CLEAR AFFILIATION) .....	88
TEST 6.3.26 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 0) .....	89
TEST 6.3.27 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 1) .....	90
TEST 6.3.28 - LINK RESET – HARD_RESET RECIVED AFTER IDENTIFY.....	91
TEST 6.3.29 - SSP_FRAMES – INTERLOCKED FRAME / NAK.....	92

*The University of New Hampshire*  
*InterOperability Laboratory*

**MODIFICATION RECORD**

- [1]September 7, 2004 (Version 0.1) DRAFT RELEASE  
David Woolf: Initial draft release
- [2]April 14, 2005 (Version 0.12) DRAFT RELEASE  
David Woolf: added STP Operations tests
- [3]July 11, 2005 (Version 0.13) DRAFT RELEASE  
David Woolf: minor edits
- [4]August 18, 2005 (Version 0.14) DRAFT RELEASE  
Leon Cyril: minor edits
- [5]September 22, 2005 (Version 1.0) DRAFT RELEASE  
Leon Cyril: added discussion for all tests.
- [6]June 5, 2006 (Version 1.1) FINAL RELEASE  
David Woolf: Modified tests 7.3.4 and 7.3.5.
- [7]July 10, 2006 (Version 1.1) FINAL RELEASE  
Michael Davidson: Added Tests 7.3.25 – 7.3.29, Removed RCC References
- [8]October 9, 2007 (Version 1.2) FINAL RELEASE  
Michael Davidson: Updated all references
- [9]July 26, 2011 (Version 1.3) FINAL RELEASE  
Joshua Beaudet: Updated all references
- [10]February 1, 2016 (Version 1.4) FINAL RELEASE  
David Woolf: Updated observable results in tests 6.1.15, 6.1.16

*The University of New Hampshire  
InterOperability Laboratory*

*The University of New Hampshire*  
*InterOperability Laboratory*

**ACKNOWLEDGMENTS**

**The University of New Hampshire would like to acknowledge the efforts of the following individuals in the development of this test suite.**

Kurtis Kofler	UNH InterOperability Laboratory (UNH-IOL)
David Woolf	UNH InterOperability Laboratory (UNH-IOL)
Leon Cyril	UNH InterOperability Laboratory (UNH-IOL)
Michael Davidson	UNH InterOperability Laboratory (UNH-IOL)
Joel Nkounkou	UNH InterOperability Laboratory (UNH-IOL)

## **INTRODUCTION**

The University of New Hampshire's InterOperability Laboratory (IOL) is an institution designed to improve the interoperability of standards based products by providing an environment where a product can be tested against other implementations of a standard. This particular suite of tests has been developed in conjunction with CATC to help implementers evaluate the functionality of their Serial Attached SCSI (SAS) products. Specifically this Test Suite is directed at verifying the Link layer of SAS Targets, Initiators, and Expanders.

These tests are designed to determine if a SAS product conforms to specifications defined in **ISO/IEC 14776-150, *Serial Attached SCSI (SAS) standard T10/1562-D, Revision 5*** (hereafter referred to as the "SAS SPL"). Successful completion of all tests contained in this suite does not guarantee that the tested device will successfully operate with other SAS products. However, when combined with satisfactory operation in the IOL's interoperability test bed, these tests provide a reasonable level of confidence that the Device Under Test (DUT) will function properly in many SAS environments.

The tests contained in this document are organized in order to simplify the identification of information related to a test, and to facilitate in the actual testing process. Tests are separated into groups, primarily in order to reduce setup time in the lab environment, however the different groups typically also tend to focus on specific aspects of device functionality. A three-number, dot-notated naming system is used to catalog the tests, where the first number always indicates the specific clause of the reference standard on which the test suite is based. The second and third numbers indicate the test's group number and test number within that group, respectively. This format allows for the addition of future tests in the appropriate groups without requiring the renumbering of the subsequent tests.

The test definitions themselves are intended to provide a high-level description of the motivation, resources, procedures, and methodologies specific to each test. Formally, each test description contains the following sections:

### **Purpose**

The purpose is a brief statement outlining what the test attempts to achieve. The test is written at the functional level.

### **References**

This section specifies all reference material *external* to the test suite, including the specific sub clauses references for the test in question, and any other references that might be helpful in understanding the test methodology and/or test results. External sources are always referenced by a bracketed number (e.g., [1]) when mentioned in the test description. Any other references in the test description that are not indicated in this manner refer to elements within the test suite document itself (e.g., "Appendix 5.A", or "Table 5.1.1-1")



### **Resource Requirements**

The requirements section specifies the test hardware and/or software needed to perform the test. This is generally expressed in terms of minimum requirements, however in some cases specific equipment manufacturer/model information may be provided.

### **Last Modification**

This specifies the date of the last modification to this test.

### **Test Setup**

The setup section describes the initial configuration of the test environment. Small changes in the configuration should not be included here, and are generally covered in the test procedure section (next).

### **Procedure**

The procedure section of the test description contains the systematic instructions for carrying out the test. It provides a cookbook approach to testing, and may be interspersed with observable results.

### **Observable Results**

This section lists the specific observables that can be examined by the tester in order to verify that the DUT is operating properly. When multiple values for an observable are possible, this section provides a short discussion on how to interpret them. The determination of a pass or fail outcome for a particular test is generally based on the successful (or unsuccessful) detection of a specific observable.

### **Possible Problems**

This section contains a description of known issues with the test procedure, which may affect test results in certain situations. It may also refer the reader to test suite appendices and/or other external sources that may provide more detail regarding these issues.

*The University of New Hampshire  
InterOperability Laboratory*

**REFERENCES**

The following document is referenced in this text:

- [1] T10/Project 1601-DT/Rev 8 – Serial Attached SCSI 1.1 – (SAS – 1.1)

## **GROUP 1: LINK LAYER TESTS FOR TARGETS**

**Overview:**

This group of tests verifies the Link Layer specifications of the SAS physical layer defined in Clause 6 of the SAS SPL, for Targets.

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.1 - LINK RESET – REPEAT PHY SEQUENCE IF NO IDENTIFY RECEIVED**

**Purpose:** To determine that the DUT will repeat the phy reset sequence if an Identify frame is not received within 1 ms of completing a phy reset sequence.

**References:**

- [1] 6.10 SAS SPL
- [2] 6.8.2 SAS SPL
- [3] 4.4.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 25, 2011

**Discussion:** After the phy reset sequence has been completed indicating the physical link is using SAS rather than SATA, each phy transmits either: a) an IDENTIFY address frame [2]; or b) a HARD\_RESET. Each phy receives an IDENTIFY address frame or a HARD\_RESET from the phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence, and an identification sequence is called a link reset sequence [3]. If a device does not receive a valid IDENTIFY address frame within 1 ms of phy reset sequence completion, it shall restart the phy reset sequence.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a Phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a Phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit valid SAS primitives or SAS Idle. The Testing Station is instructed to not transmit an Identify Address Frame.

**Observable Results:** Verify that 1 ms after the DUT completed a phy Reset Sequence with the DUT, the DUT initiated another phy reset sequence by transmitting COMINIT.

**Possible Problems:** None.

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.2 - LINK RESET – IGNORE ADDITIONAL IDENTIFY RECEIVED**

**Purpose:** To determine that the DUT will ignore an Identify frame received after a valid Identify has already been received.

**References:**

- [1] 6.10 SAS SPL
- [2] 6.8.2 SAS SPL
- [3] 4.4.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 25, 2011

**Discussion:** After the phy reset sequence has been completed indicating the physical link is using SAS rather than SATA, each phy transmits either: a) an IDENTIFY address frame [2]; or b) a HARD\_RESET. Each phy receives an IDENTIFY address frame or a HARD\_RESET from the phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence, and an identification sequence is called a link reset sequence [3]. If a device receives an additional IDENTIFY address frame after receiving the first one, without an intervening phy reset sequence, it shall ignore the additional IDENTIFY address frame.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit valid SAS primitives or SAS Idle. The Testing Station is instructed to transmit 2 valid Identify Address Frames each with a different SAS address.

**Observable Results:** Verify that the DUT ignores the second received Identify Address Frame and does not transmit COMINIT. Also, verify that the DUT does not attempt to open a connection to the address in the second Identify Address frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.3 - LINK RESET – HARD\_RESET RECEIVED**

**Purpose:** To determine that the DUT properly handles a received HARD\_RESET.

**References:**

- [1] 6.2.5.3 SAS SPL
- [2] 6.10.1 SAS SPL
- [3] 4.4 SAS SPL
- [4] 4.4.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 25, 2011

**Discussion:** HARD\_RESET is used to force a phy to generate a hard reset to its port. This primitive is only valid after the phy reset sequence without an intervening identification sequence [3] and shall be ignored at other times. If a phy receives a HARD\_RESET, it shall be considered a reset event and cause a hard reset [4] of the port containing that phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit 6 consecutive HARD\_RESET primitives. This must occur before the Identify sequence is complete.

**Observable Results:** Verify that the DUT transmits COMINIT upon receiving the HARD\_RESET primitives from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.4 - CONNECTIONS – OPEN\_ACCEPT**

**Purpose:** To determine that the DUT properly transmits RRDY after receiving OPEN\_ACCEPT.

**References:**

[1] 6.13.2.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 25, 2011

**Discussion:** The OPEN address frame is used to request that a connection be opened. AIP, OPEN\_ACCEPT and OPEN\_REJECT are the responses to an OPEN address frame. An OPEN\_ACCEPT indicates that the connection request was accepted. This is sent by the destination phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to transmit an Open Address frame to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.

**Observable Results:** Verify that the DUT transmitted RRDY no more than 1 ms after transmitted OPEN\_ACCEPT.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.5 - CONNECTIONS – 2 SOAF RECEIVED**

**Purpose:** To determine that the DUT properly handles received 2 SOAF primitives.

**References:**

[1] 6.8.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 25, 2011

**Discussion:** A complete OPEN ADDRESS frame will start with an SOAF primitive and end with an EOAF primitive. A SAS device must be capable of detecting an OPEN ADDRESS frame in a stream of SAS primitives.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to transmit the following to the DUT
  - a. SOAF primitive
  - b. the first 4 dwords of an OpenAddress frame
  - c. complete Open Address frame starting with SOAF and using the SAS address of the DUT.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_ACCEPT.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.6 - CONNECTIONS - OPEN\_REJECT CONNECTION RATE NOT SUPPORTED**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

- [1] 6.8.1 SAS SPL
- [2] 6.8.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (CONNECTION RATE NOT SUPPORTED) occurs when the requested connection rate is not supported on some physical link on the pathway between the source phy and destination phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to transmit an OpenAddress frame to the DUT with an invalid connection rate.
3. Wait for the DUT to transmit OPEN\_REJECT (Connection rate not supported).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Connection rate not supported).

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.7 - CONNECTIONS - OPEN\_REJECT PROTOCOL NOT SUPPORTED**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

[1] 6.8.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (PROTOCOL NOT SUPPORTED) occurs when the device with destination SAS address exists but the destination device does not support the requested initiator/target role, protocol, initiator connection tag, or features.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to transmit an OpenAddress frame to the DUT with an invalid initiator or target role.
3. Wait for the DUT to transmit OPEN\_REJECT (Protocol not supported).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Protocol not supported).

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.8 - CONNECTIONS - OPEN\_REJECT WRONG DESTINATION**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

[1] 6.8.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (WRONG DESTINATION) occurs when the destination SAS address does not match the SAS address of the SAS port to which the connection request was delivered.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to transmit an OpenAddress frame to the DUT with an incorrect destination SAS Address.
3. Wait for the DUT to transmit OPEN\_REJECT (Wrong Destination).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Wrong Destination).

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.9 - CONNECTIONS - OPEN\_REJECT RETRY**

**Purpose:** To determine that the DUT handles extra connection requests properly.

**References:**

[1] 6.8.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (RETRY) occurs when the device with destination SAS address exists but is not able to accept connections.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to continuously transmit OpenAddress frames to the DUT with a correct destination SAS Address.
3. Wait for the DUT to transmit OPEN\_REJECT (Retry).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Retry). This item may not be testable for some devices.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.10 - SSP\_FRAMES – INTERLOCKED FRAME**

**Purpose:** To determine that the DUT handles interlocked frames properly.

**References:**

- [1] 6.17.3 SAS SPL
- [2] 6.16.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** Before transmitting an interlocked frame, an SSP phy shall wait for all SSP frames to be acknowledged with ACK or NAK, even if credit is available. After transmitting an interlocked frame, an SSP phy shall not transmit another SSP frame until it has been acknowledged with ACK or NAK, even if credit is available.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open a SSP connection to the DUT and transmit a SCSI\_INQUIRY command.
3. Wait for the DUT to transmit ACK followed by SCSI Response.

**Observable Results:** Verify that upon receiving the INQUIRY command, the DUT transmitted ACK before transmitting any other frame to the Testing Station.

**Possible Problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.1.11 - SSP\_FRAMES - NO ACK FOR INTERLOCKED FRAME**

**Purpose:** To determine that the DUT responds properly when no ACK or NAK is received after transmitting an interlocked frame.

**References:**

- [1] 6.17.3 SAS SPL
- [2] 6.16.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** Before transmitting an interlocked frame, an SSP phy shall wait for all SSP frames to be acknowledged with ACK or NAK, even if credit is available. After transmitting an interlocked frame, an SSP phy shall not transmit another SSP frame until it has been acknowledged with ACK or NAK, even if credit is available.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open a SSP connection to the DUT and transmit a SCSI\_INQUIRY command.
3. Wait for the DUT to transmit ACK followed by SCSI Response.
4. The Testing Station is instructed to not transmit ACK or NAK to the SCSI Response frame.
5. The Testing Station is instructed to transmit a second SCSI\_INQUIRY command with a different TAG than the first SCSI INQUIRY command.

**Observable Results:** Verify that the DUT aborts the first INQUIRY command with DONE (ACK/NAK TIMEOUT) after the ACK/NAK timeout of 1 ms.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.12 - SSP\_FRAMES – MULTIPLE ACKS**

**Purpose:** To determine that the DUT properly transmits ACK within 1 ms of receiving a frame requiring an ACK response.

**References:**

- [1] 6.17.3 SAS SPL
- [2] 6.16.5 SAS SPL
- [3] 6.17.8.11 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer, SAS Initiator, SAS Target, Software capable of generating SCSI traffic from the SAS Initiator. In order to minimize physical layer problems, the Initiator, Target, and Analyzer should be connected with near-ideal channels.

**Last Modification:** July 26, 2011

**Discussion:** Receiving SSP phys shall acknowledge SSP frames within 1 ms if not discarded as described in [3] with either a positive acknowledgement (ACK) or a negative acknowledgement (NAK). ACK means the SSP frame was received into a frame buffer without errors. NAK (CRC ERROR) means the SSP frame was received with a CRC error, an invalid dword, or an ERROR primitive.

**Test Setup:** The SAS Initiator and Target are connected through the SAS Analyzer.

**Test Procedure:**

1. Using the generation software, cause the SAS Initiator to begin a series of 1000 64 kB READ operations on the SAS Target. Capture this on the SAS Analyzer.

**Observable Results:** If the DUT is an initiator, search the Analyzer capture of the target transmissions for the DONE (ACK/NAK TIMEOUT) primitive. If the DUT is a target, search the Analyzer capture of the initiator transmissions for the DONE (ACK/NAK TIMEOUT) primitive. If any of these primitives are found, it indicates that the DUT did not transmit ACK within the 1 ms necessary after receiving frame requiring ACK or NAK.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.13 - SSP FRAMES - RRDY**

**Purpose:** To determine that the DUT properly grants credit to transit frames using RRDY.

**References:**

[1] 6.17.8.4 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** An SSP phy uses RRDY to grant credit for permission for the other SSP phy in the connection to transmit frames. Each RRDY increments credit by one frame. Frame transmission decrements credit by one frame. Credit of zero frames is established at the beginning of each connection.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.

**Observable Results:** Verify that the DUT transmitted RRDY when an SSP connection was opened.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.14 - CLOSING SSP CONNECTIONS – DONE (NORMAL)**

**Purpose:** To determine that the DUT properly responds when DONE is received.

**References:**

- [1] 6.17.8.8 SAS SPL
- [2] 6.2.7.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** DONE shall be exchanged prior to closing an SSP connection [2]. There are several versions of the DONE primitive indicating additional information about why the SSP connection is being closed: DONE (NORMAL) specifies normal completion; the transmitter has no more SSP frames to transmit.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the Testing Station.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.
4. The Testing Station is instructed to transmit DONE (NORMAL).

**Observable Results:** Verify that the DUT transmitted DONE (NORMAL) within 1 ms of receiving DONE (NORMAL) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.15 - CLOSING SSP CONNECTIONS – DONE (ACK/NAK TIMEOUT)**

**Purpose:** To determine that the DUT properly responds when ACK or NAK has not been received for a transmitted frame.

**References:**

- [1] 6.17.8.7 SAS SPL
- [2] 6.2.7.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** February 1, 2016

**Discussion:** DONE shall be exchanged prior to closing an SSP connection [2]. There are several versions of the DONE primitive indicating additional information about why the SSP connection is being closed: DONE (ACK/NAK TIMEOUT) specifies that the transmitter transmitted an SSP frame but did not receive the corresponding ACK or NAK within 1 ms. As a result, the ACK/NAK count is not balanced and the transmitter is going to transmit a BREAK in 1 ms unless the recipient replies with DONE and the connection is closed.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the Testing Station.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY. The Testing Station is instructed to transmit a SCSI INQUIRY command to the DUT, then close the connection.
4. Wait for the DUT to open an SSP Connection to the Testing Station. Allow the DUT to transmit a SCSI Response. The Testing Station is instructed to not transmit ACK or NAK.

**Observable Results:** Verify that the DUT transmitted DONE (ACK/NAK TIMEOUT) no sooner than 1 ms after transmitting the Command or Response frame.

**Possible Problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.1.16 - CLOSING SSP CONNECTIONS – DONE (CREDIT TIMEOUT)**

**Purpose:** To determine that the DUT properly responds when RRDY has not been received for an impending transaction.

**References:**

- [1] 6.17.8 SAS SPL
- [2] 6.2.7.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** February 1, 2016

**Discussion:** DONE shall be exchanged prior to closing an SSP connection [2]. There are several versions of the DONE primitive indicating additional information about why the SSP connection is being closed: DONE (CREDIT TIMEOUT) specifies that the transmitter still has SSP frames to transmit but did not receive an RRDY granting frame credit within 1 ms, or the transmitter has received a CREDIT\_BLOCKED and has consumed all RRDYs received.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY. The Testing Station is instructed to transmit a SCSI INQUIRY command to the DUT, then close the connection.
4. Wait for the DUT to open an SSP connection to the Testing Station. The Testing Station is instructed to not transmit RRDY to grant credit for the DUT to transmit a SCSI Response to the Testing Station.

**Observable Results:** Verify that the DUT transmitted DONE (CREDIT TIMEOUT) no sooner than 1 ms after opening the SSP connection.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.17 - CLOSING SSP CONNECTIONS – CLOSE (NORMAL)**

**Purpose:** To determine that the DUT properly responds when CLOSE (NORMAL) is received.

**References:**

- [1] 6.13.7 SAS SPL
- [2] 6.17.8.1 SAS SPL
- [3] 6.18.6 SAS SPL
- [4] 6.19.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [2] for details on closing SSP connections, [3] for details on closing STP connections, and [4] for details on closing SMP connections. After transmitting CLOSE, the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (NORMAL) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (NORMAL) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.18 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY TESTING STATION**

**Purpose:** To determine that the DUT properly responds when BREAK is received.

**References:**

[1] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** BREAK may be used to abort a connection request. The source phy shall transmit a BREAK after the Open Timeout timer expires or if it chooses to abort its request for any other reason. After transmitting BREAK, the source phy shall initialize a Break Timeout timer to 1 ms and start the Break Timeout timer.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to transmit an OpenAddress frame to the DUT, followed by BREAK.

**Observable Results:** Verify that the DUT transmitted BREAK or BREAK\_REPLY within 1 ms of receiving BREAK from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.19 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY DUT**

**Purpose:** To determine that the DUT properly sources BREAK when required.

**References:**

[1] 6.13.7 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** In addition to aborting a connection request, BREAK may also be used to break a connection, in cases where CLOSE is not available. After transmitting BREAK, the originating phy shall ignore all incoming dwords except for BREAKs.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY. The Testing Station is instructed to transmit a SCSI INQUIRY command to the DUT, then close the connection.
4. Wait for the DUT to open an SSP connection to the Testing Station. Allow the DUT to transmit a SCSI Response to the Testing Station.
5. When the DUT transmits CLOSE (NORMAL) to close the connection, the Testing Station is instructed to not transmit CLOSE (NORMAL) in response.

**Observable Results:** Verify that the DUT transmitted BREAK within 1 ms of transmitting CLOSE (NORMAL) to the Testing Station.

**Possible problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.1.20 - CONNECTIONS THROUGH EXPANDERS – OPEN SSP TARGET**

**Purpose:** To determine that the DUT, can properly open a connection through an expander to transmit data or status.

**References:**

[1] 6.13.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** A connection is opened between a SAS initiator port and a SAS target port before communication begins. A connection is established between one SAS initiator phy in the SAS initiator port and one SAS target phy in the SAS target port. SSP initiator ports open SSP connections to transmit SCSI commands, task management functions, or transfer data. SSP target ports open SSP connections to transfer data or transmit status. SMP initiator ports open SMP connections to transmit SMP requests and receive SMP responses. STP initiator ports and STP target ports open STP connections to transmit SATA frames.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Edge Expander.
2. The Testing Station is instructed to open an SSP connection to the DUT from another address, imitating a SAS Initiator.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY. The Testing Station is instructed to transmit a SCSI INQUIRY command to the DUT, then close the connection.
4. Wait for the DUT to open an SSP connection to the Testing Station to the same address that sourced the SCSI INQUIRY command.
5. Before transmitting OPEN\_ACCEPT the Testing Station is instructed to transmit AIP(NORMAL) followed by AIP(WAITING ON DEVICE)
6. Allow the DUT to transmit a SCSI Data frame to the Testing Station.
7. When the DUT transmits CLOSE (NORMAL) to close the connection, the Testing Station is instructed to transmit CLOSE (NORMAL) in response.
8. Wait for the DUT to open an SSP connection to the Testing Station to the same address that sourced the SCSI INQUIRY command. Allow the DUT to transmit a SCSI Response frame to the Testing Station.
9. When the DUT transmits CLOSE (NORMAL) to close the connection, the Testing Station is instructed to transmit CLOSE (NORMAL) in response.

**Observable Results:** Verify that the DUT successfully transmitted both SCSI Response and Data frames.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.21 - BREAK – OPENTIMEOUT TIMER SSP TARGET**

**Purpose:** To determine that the DUT, an SSP target transmits BREAK after the OpenTimeout Timer expires.

**References:**

- [1] 6.13.2.1 SAS SPL
- [2] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** To make a connection request, the source port shall transmit an OPEN address frame through an available phy. The source phy shall transmit idle dwords after the OPEN address frame until it receives a response or aborts the connection request with BREAK. After transmitting an OPEN address frame, the source phy shall initialize and start a 1 ms Open Timeout timer. If the Open Timeout timer expires before a connection response is received, the source phy may assume the destination port does not exist and shall transmit BREAK to abort the connection request [2].

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Edge Expander.
2. The Testing Station is instructed to open an SSP connection to the DUT from another address, imitating a SAS Initiator.
3. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY. The Testing Station is instructed to transmit a SCSI INQUIRY command to the DUT, then close the connection.
4. Wait for the DUT to open an SSP connection to the Testing Station to the same address that sourced the SCSI INQUIRY command.
5. The Testing Station is instructed to wait for the DUT to transmit an OpenAddress frame for an SSP connection. The Testing Station is instructed to not respond to the OpenAddress frame with OPEN\_ACCEPT or OPEN\_REJECT.

**Observable Results:** Verify that 1 ms after transmitting the OpenAddress frame, the DUT transmitted BREAK.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.22 - CLOSING SSP CONNECTIONS – CLOSE (CLEAR AFFILIATION)**

**Purpose** – To determine that the DUT properly responds when CLOSE (CLEAR AFFILIATION) is received.

**Reference:**

- [1] 6.2.6.5 SAS SPL
- [2] 6.13.6 SAS SPL
- [3] 6.17.8 SAS SPL
- [4] 6.18.7 SAS SPL
- [5] 6.19.5 SAS SPL
- [6] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [3] for details on closing SSP connections, [4] for details on closing STP connections, and [5] for details on closing SMP connections. After transmitting CLOSE (CLEAR AFFILIATION), the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

CLOSE (CLEAR AFFILIATION) closes an open STP connection and clears the affiliation [1][6]. CLOSE (CLEAR AFFILIATION) is processed the same as CLOSE (NORMAL) if:

- a) the connection is not an STP connection;
- b) the connection is an STP connection, but affiliations are not implemented by the STP target port; or
- c) the connection is an STP connection, but an affiliation is not present.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

- 3. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
- 4. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (CLEAR AFFILIATION) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (CLEAR AFFILIATION) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.23 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 0)**

**Purpose** – To determine that the DUT properly responds when CLOSE (RESERVED 0) is received.

**Reference:**

- [1] 6.2.6.5 SAS SPL
- [2] 6.13.6 SAS SPL
- [3] 6.17.8 SAS SPL
- [4] 6.18.7 SAS SPL
- [5] 6.19.5 SAS SPL
- [6] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [3] for details on closing SSP connections, [4] for details on closing STP connections, and [5] for details on closing SMP connections. After transmitting CLOSE (RESERVED 0), the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

CLOSE (RESERVED 0) closes an open STP connection and clears the affiliation [1][6]. CLOSE (RESERVED 0) is processed the same as CLOSE (NORMAL) if:

- a) the connection is not an STP connection;
- b) the connection is an STP connection, but affiliations are not implemented by the STP target port; or
- c) the connection is an STP connection, but an affiliation is not present.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (RESERVED 0) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (RESERVED 0) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.24 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 1)**

**Purpose** – To determine that the DUT properly responds when CLOSE (RESERVED 1) is received.

**Reference:**

- [1] 6.2.6.5 SAS SPL
- [2] 6.13.6 SAS SPL
- [3] 6.17.8 SAS SPL
- [4] 6.18.7 SAS SPL
- [5] 6.19.5 SAS SPL
- [6] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [3] for details on closing SSP connections, [4] for details on closing STP connections, and [5] for details on closing SMP connections. After transmitting CLOSE (RESERVED 1), the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

CLOSE (RESERVED 0) closes an open STP connection and clears the affiliation [1][6]. CLOSE (RESERVED 1) is processed the same as CLOSE (NORMAL) if:

- a) the connection is not an STP connection;
- b) the connection is an STP connection, but affiliations are not implemented by the STP target port; or
- c) the connection is an STP connection, but an affiliation is not present.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (RESERVED 1) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (RESERVED 1) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.25 - LINK RESET – HARD\_RESET RECEIVED AFTER IDENTIFY**

**Purpose:** To determine that the DUT properly handles a received HARD\_RESET.

**References:**

- [1] 6.2.5.3 SAS SPL
- [2] 6.10.1 SAS SPL
- [3] 4.4 SAS SPL
- [4] 4.4.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** HARD\_RESET is used to force a phy to generate a hard reset to its port. This primitive is only valid after the phy reset sequence without an intervening identification sequence [3] and shall be ignored at other times. If a phy receives a HARD\_RESET, it shall be considered a reset event and cause a hard reset [4] of the port containing that phy.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit 6 consecutive HARD\_RESET primitives. This must occur after the Identify sequence is complete.

**Observable Results:** Verify that the DUT does not transmit COMINIT upon receiving the HARD\_RESET primitives from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.1.26 - SSP\_FRAMES – INTERLOCKED FRAME / NAK**

**Purpose:** To determine that the DUT handles interlocked frames with CRC errors properly.

**References:**

- [1] 6.17.3 SAS SPL
- [2] 6.16.5 SAS SPL
- [3] 6.2.6.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** Before transmitting an interlocked frame, an SSP phy shall wait for all SSP frames to be acknowledged with ACK or NAK, even if credit is available. After transmitting an interlocked frame, an SSP phy shall not transmit another SSP frame until it has been acknowledged with ACK or NAK, even if credit is available. A receiving SSP phy shall respond to a SSP frame with NAK (CRC ERROR) if the SSP frame was received with a bad CRC.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open a SSP connection to the DUT and transmit a SCSI\_INQUIRY command with an invalid CRC.
3. Wait for the DUT to transmit NAK (CRC ERROR).

**Observable Results:** Verify that upon receiving the INQUIRY command with an invalid CRC, the DUT transmitted NAK (CRC ERROR) before transmitting any other frame to the Testing Station.

**Possible Problems:** None

## **GROUP 2: LINK LAYER TESTS FOR INITIATORS**

**Overview:**

This group of tests verifies the Link Layer specifications of the SAS physical layer defined in Clause 6 of the SAS SPL, for Initiators.

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.1 - LINK RESET – REPEAT PHY SEQUENCE IF NO IDENTIFY RECEIVED**

**Purpose:** To determine that the DUT will repeat the phy reset sequence if an Identify frame is not received within 1 ms of completing a phy reset sequence.

**References:**

- [1] 6.10.1 SAS SPL
- [2] 6.8.2 SAS SPL
- [3] 4.4.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** After the phy reset sequence has been completed indicating the physical link is using SAS rather than SATA, each phy transmits either: a) an IDENTIFY address frame [2]; or b) a HARD\_RESET. Each phy receives an IDENTIFY address frame or a HARD\_RESET from the phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence, and an identification sequence is called a link reset sequence [3]. If a device does not receive a valid IDENTIFY address frame within 1 ms of phy reset sequence completion, it shall restart the phy reset sequence.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit valid SAS primitives or SAS Idle. The Testing Station is instructed to not transmit an Identify Address Frame.

**Observable Results:** Verify that 1 ms after the DUT completed a phy Reset Sequence with the Testing Station, the DUT initiated another phy reset sequence by transmitting COMINIT.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.2 - LINK RESET – IGNORE ADDITIONAL IDENTIFY RECEIVED**

**Purpose:** To determine that the DUT will ignore an Identify frame received after a valid Identify has already been received.

**References:**

- [1] 6.10.1 SAS SPL
- [2] 6.8.2 SAS SPL
- [3] 4.4.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** After the phy reset sequence has been completed indicating the physical link is using SAS rather than SATA, each phy transmits either: a) an IDENTIFY address frame [2]; or b) a HARD\_RESET. Each phy receives an IDENTIFY address frame or a HARD\_RESET from the phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence, and an identification sequence is called a link reset sequence [3]. If a device receives an additional IDENTIFY address frame after receiving the first one, without an intervening phy reset sequence, it shall ignore the additional IDENTIFY address frame.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit valid SAS primitives or SAS Idle. The Testing Station is instructed to transmit 2 valid Identify Address Frames each with a different SAS address.

**Observable Results:** Verify that the DUT ignores the second received Identify Address Frame and does not transmit COMINIT. Also, verify that the DUT does not attempt to open a connection to the address in the second Identify Address frame.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.3 - LINK RESET – HARD\_RESET RECEIVED**

**Purpose:** To determine that the DUT properly handles a received HARD\_RESET.

**References:**

- [1] 6.2.5.3 SAS SPL
- [2] 6.10.1 SAS SPL
- [3] 4.4 SAS SPL
- [4] 4.4.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** HARD\_RESET is used to force a phy to generate a hard reset to its port. This primitive is only valid after the phy reset sequence without an intervening identification sequence [3] and shall be ignored at other times. If a phy receives a HARD\_RESET, it shall be considered a reset event and cause a hard reset [4] of the port containing that phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit 6 consecutive HARD\_RESET primitives. This must occur before the Identify sequence is complete.

**Observable Results:** Verify that the DUT transmits COMINIT upon receiving the HARD\_RESET primitives from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.4 - LINK RESET – SMP INITIATOR TO PERFORM DISCOVER**

**Purpose:** To determine that the DUT, which is an SMP Initiator, properly performs the discovery process.

**References:**

- [1] 4.6.7.4 SAS SPL
- [2] 8.4.3.4 SAS SPL
- [3] 8.4.3.10 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** Management application clients direct an SMP initiator port to request SMP functions from an SMP target port. If an expander device is attached, the management application client shall use the SMP REPORT GENERAL function [3] to determine how many phys are in the expander device and then use the SMP DISCOVER function [4] to determine what is attached to each expander phy (e.g., the device type, SAS address, and supported protocol(s)).

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Edge Expander.
4. The DUT is expected to transmit a SMP Request REPORT GENERAL, the Testing Station is instructed to respond with SMP Response REPORT GENERAL indicating a phy Count of 1. The DUT is expected to transmit an SMP Request DISCOVER to the Testing Station with a phy ID of 0x00.
5. The Testing Station is instructed to transmit an SMP Request DISCOVER indicating the presence of an SSP Target.

**Observable Results:** Verify that the DUT transmits both an SMP Request REPORT GENERAL and an SMP Request DISCOVER to the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.5 - LINK RESET – EDGE EXPANDER TRANSMITS OPEN\_REJECT**

**Purpose:** To determine that the DUT, properly handles receiving OPEN\_REJECT from an Edge Expander.

**References:**

[1] 6.10 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** After completing the identification sequence on a phy and completing internal initialization, the ECM within an edge expander device shall be capable of routing connection requests through that phy. The expander device may return OPEN\_REJECT (NO DESTINATION) until it is ready for connection requests.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Edge Expander.
4. The DUT is expected to transmit a SMP Request REPORT GENERAL, the Testing Station is instructed to respond with SMP Response REPORT GENERAL indicating a phy Count of 1. The DUT is expected to transmit an SMP Request DISCOVER to the Testing Station with a phy ID of 0x00.
5. The Testing Station is instructed to transmit an SMP Request DISCOVER indicating the presence of an SSP Target.

**Observable Results:** Verify that the DUT retries the OPEN Address frame for the connection request. Also, verify that the DUT does not attempt phy Reset upon receiving the OPEN\_REJECT primitive.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.6 - CONNECTIONS – OPEN\_ACCEPT**

**Purpose:** To determine that the DUT properly transmits RRDY after receiving OPEN\_ACCEPT.

**References:**

[1] 6.13.2.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** The OPEN address frame is used to request that a connection be opened. AIP, OPEN\_ACCEPT and OPEN\_REJECT are the responses to an OPEN address frame. An OPEN\_ACCEPT indicates that the connection request was accepted. This is sent by the destination phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to transmit an Open Address frame to the Testing Station. The Testing Station is instructed to transmit OPEN\_ACCEPT followed by RRDY. Allow the DUT to transmit a frame and close the connection.
3. The Testing Station is instructed to transmit an Open Address frame to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.

**Observable Results:** Verify that the DUT transmitted RRDY no more than 1 ms after the Testing Station transmitted OPEN\_ACCEPT.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.7 - CONNECTIONS – 2 DIFFERENT SOAF RECEIVED**

**Purpose:** To determine that the DUT properly handles received 2 SOAF primitives.

**References:**

[1] 6.8 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** A complete OPEN ADDRESS frame will start with an SOAF primitive and end with an EOAF primitive. A SAS device must be capable of detecting an OPEN ADDRESS frame in a stream of SAS primitives.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to transmit an Open Address frame to the Testing Station. The Testing Station is instructed to transmit OPEN\_ACCEPT followed by RRDY. Allow the DUT to transmit a frame and close the connection.
3. The Testing Station is instructed to transmit the following to the DUT
  - a. SOAF primitive
  - b. the first 4 dwords of an OpenAddress frame
  - c. complete Open Address frame starting with SOAF.
4. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_ACCEPT.

**Possible Problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.2.8 - CONNECTIONS - OPEN\_REJECT CONNECTION RATE NOT SUPPORTED**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

- [1] 6.8 SAS SPL
- [2] 6.8.3 SAS SPL
- [3] Table 149 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (CONNECTION RATE NOT SUPPORTED) occurs when the requested connection rate is not supported on some physical link on the pathway between the source phy and destination phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to transmit an Open Address frame to the Testing Station. The Testing Station is instructed to transmit OPEN\_ACCEPT followed by RRDY. Allow the DUT to transmit a frame and close the connection.
3. The Testing Station is instructed to transmit an OpenAddress frame to the DUT with an invalid connection rate.
4. Wait for the DUT to transmit OPEN\_REJECT (Connection rate not supported).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Connection rate not supported).

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.9 - CONNECTIONS - OPEN\_REJECT PROTOCOL NOT SUPPORTED**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

[1] 6.8 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (PROTOCOL NOT SUPPORTED) occurs when the device with destination SAS address exists but the destination device does not support the requested initiator/target role, protocol, initiator connection tag, or features.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to transmit an Open Address frame to the Testing Station. The Testing Station is instructed to transmit OPEN\_ACCEPT followed by RRDY. Allow the DUT to transmit a frame and close the connection.
3. The Testing Station is instructed to transmit an OpenAddress frame to the DUT with an invalid initiator or target role.
4. Wait for the DUT to transmit OPEN\_REJECT (Protocol not supported).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Protocol not supported).

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.10 - CONNECTIONS - OPEN\_REJECT WRONG DESTINATION**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

[1] 6.8 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (WRONG DESTINATION) occurs when the destination SAS address does not match the SAS address of the SAS port to which the connection request was delivered.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to transmit an Open Address frame to the Testing Station. The Testing Station is instructed to transmit OPEN\_ACCEPT followed by RRDY. Allow the DUT to transmit a frame and close the connection.
3. The Testing Station is instructed to transmit an OpenAddress frame to the DUT with an incorrect destination SAS Address.
4. Wait for the DUT to transmit OPEN\_REJECT (Wrong Destination).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Wrong Destination).

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.11 - CONNECTIONS - OPEN\_REJECT RETRY**

**Purpose:** To determine that the DUT handles extra connection requests properly.

**References:**

[1] 6.8 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (RETRY) occurs when the device with destination SAS address exists but is not able to accept connections.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to transmit an Open Address frame to the Testing Station. The Testing Station is instructed to transmit OPEN\_ACCEPT followed by RRDY. Allow the DUT to transmit a frame and close the connection.
3. The Testing Station is instructed to continuously transmit OpenAddress frames to the DUT with a correct destination SAS Address.
4. Wait for the DUT to transmit OPEN\_REJECT (Retry).

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Retry). This item may not be testable for some devices.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.12 - SSP\_FRAMES – MULTIPLE ACKs**

**Purpose:** To determine that the DUT properly transmits ACK within 1 ms of receiving a frame requiring an ACK response.

**References:**

- [1] 6.17.3 SAS SPL
- [2] 6.16.5 SAS SPL
- [3] 6.17.8.7 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer, SAS Initiator, SAS Target, Software capable of generating SCSI traffic from the SAS Initiator. In order to minimize physical layer problems, the Initiator, Target, and Analyzer should be connected with near-ideal channels.

**Last Modification:** July 26, 2011

**Discussion:** Receiving SSP phys shall acknowledge SSP frames within 1 ms if not discarded as described in [3] with either a positive acknowledgement (ACK) or a negative acknowledgement (NAK). ACK means the SSP frame was received into a frame buffer without errors. NAK (CRC ERROR) means the SSP frame was received with a CRC error, an invalid dword, or an ERROR primitive.

**Test Setup:** The SAS Initiator and Target are connected through the SAS Analyzer.

**Test Procedure:**

1. Using the generation software, cause the SAS Initiator to begin a series of 1000 64 kB READ operations on the SAS Target. Capture this on the SAS Analyzer.

**Observable Results:** Search the Analyzer capture of the target transmissions for the DONE (ACK/NAK TIMEOUT) primitive. If any of these primitives are found, it indicates that the DUT did not transmit ACK within the 1 ms necessary after receiving frame requiring ACK or NAK.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.13 - SSP FRAMES - RRDY**

**Purpose:** To determine that the DUT properly grants credit to transit frames using RRDY.

**References:**

[1] 6.17.4 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** An SSP phy uses RRDY to grant credit for permission for the other SSP phy in the connection to transmit frames. Each RRDY increments credit by one frame. Frame transmission decrements credit by one frame. Credit of zero frames is established at the beginning of each connection.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to open an SSP connection to the Testing Station. Allow the DUT to transmit a SCSI Command to the Testing Station, then close the connection.
3. The Testing Station is instructed to transmit an OpenAddress frames to the DUT with a correct destination SAS Address.
4. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.

**Observable Results:** Verify that the DUT transmitted RRDY when an SSP connection was opened.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.14 - COSING SSP CONNECTIONS – DONE (NORMAL)**

**Purpose:** To determine that the DUT properly responds when DONE is received.

**References:**

- [1] 6.17.8.1 SAS SPL
- [2] 6.2.7.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** DONE shall be exchanged prior to closing an SSP connection [2]. There are several versions of the DONE primitive indicating additional information about why the SSP connection is being closed: DONE (NORMAL) specifies normal completion; the transmitter has no more SSP frames to transmit.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to open an SSP connection to the Testing Station. Allow the DUT to transmit a SCSI Command to the Testing Station, then close the connection.
3. The Testing Station is instructed to transmit an OpenAddress frames to the DUT with a correct destination SAS Address.
4. Wait for the DUT to transmit OPEN\_ACCEPT and RRDY.
5. The Testing Station is instructed to transmit DONE (NORMAL).

**Observable Results:** Verify that the DUT transmitted DONE (NORMAL) within 1 ms of receiving DONE (NORMAL) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.15 - CLOSING SSP CONNECTIONS – DONE (ACK/NAK TIMEOUT)**

**Purpose:** To determine that the DUT properly responds when ACK or NAK has not been received for a transmitted frame.

**References:**

- [1] 6.17.8.1 SAS SPL
- [2] 6.2.7.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** DONE shall be exchanged prior to closing an SSP connection [2]. There are several versions of the DONE primitive indicating additional information about why the SSP connection is being closed: DONE (ACK/NAK TIMEOUT) specifies that the transmitter transmitted an SSP frame but did not receive the corresponding ACK or NAK within 1 ms. As a result, the ACK/NAK count is not balanced and the transmitter is going to transmit a BREAK in 1 ms unless the recipient replies with DONE and the connection is closed.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to open an SSP connection to the Testing Station. Allow the DUT to transmit a SCSI Command to the Testing Station. The Testing Station is instructed to not transmit ACK or NAK.

**Observable Results:** Verify that the DUT transmitted DONE (ACK/NAK TIMEOUT) within 1 ms of transmitting the Command or Response frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.16 - CLOSING SSP CONNECTIONS – DONE (CREDIT TIMEOUT)**

**Purpose:** To determine that the DUT properly responds when RRDY has not been received for an impending transaction.

**References:**

- [1] 6.17.8.1 SAS SPL
- [2] 6.2.7.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** DONE shall be exchanged prior to closing an SSP connection [2]. There are several versions of the DONE primitive indicating additional information about why the SSP connection is being closed: DONE (CREDIT TIMEOUT) specifies that the transmitter still has SSP frames to transmit but did not receive an RRDY granting frame credit within 1ms, or the transmitter has received a CREDIT\_BLOCKED and has consumed all RRDYs received.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to open an SSP connection to the Testing Station. The Testing Station is instructed to not transmit RRDY to grant credit for the DUT to transmit a SCSI Command to the Testing Station.

**Observable Results:** Verify that the DUT transmitted DONE (CREDIT TIMEOUT) within 1 ms of opening the SSP connection.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.17 - CLOSING SSP CONNECTIONS – CLOSE (NORMAL)**

**Purpose:** To determine that the DUT properly responds when CLOSE (NORMAL) is received.

**References:**

- [1] 6.17.8.1 SAS SPL
- [2] 6.13.7 SAS SPL
- [3] 6.18.6 SAS SPL
- [4] 6.19.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [1] for details on closing SSP connections, [3] for details on closing STP connections and [4] for details on closing SMP connections. After transmitting CLOSE, the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to open an SSP connection to the Testing Station. Allow the DUT to transmit a SCSI Command to the Testing Station, then close the connection.
3. The Testing Station is instructed to transmit an OpenAddress frame to the DUT. Wait for OPEN\_ACCEPT from the DUT, then transmit CLOSE (NORMAL)

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (NORMAL) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.18 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY TESTING STATION**

**Purpose:** To determine that the DUT properly responds when BREAK is received.

**References:**

[1] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** BREAK may be used to abort a connection request. The source phy shall transmit a BREAK after the Open Timeout timer expires or if it chooses to abort its request for any other reason. After transmitting BREAK, the source phy shall initialize a Break Timeout timer to 1 ms and start the Break Timeout timer.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to open an SSP connection to the Testing Station. Allow the DUT to transmit a SCSI Command to the Testing Station, then close the connection.
3. The Testing Station is instructed to transmit an OpenAddress frame to the DUT, followed by BREAK.

**Observable Results:** Verify that the DUT transmitted BREAK or BREAK\_REPLY within 1 ms of receiving BREAK from the Testing Station.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.19 - CLOSING SSP CONNECTIONS – BREAK SOURCED BY DUT**

**Purpose:** To determine that the DUT properly sources BREAK when required.

**References:**

[1] 6.13.7 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** In addition to aborting a connection request, BREAK may also be used to break a connection, in cases where CLOSE is not available. After transmitting BREAK, the originating phy shall ignore all incoming dwords except for BREAKs.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is a target.
2. Wait for the DUT to open an SSP connection to the Testing Station. Allow the DUT to transmit a SCSI Command to the Testing Station.
3. When the DUT transmits CLOSE (NORMAL) to close the connection, the Testing Station is instructed to not transmit CLOSE (NORMAL) in response.

**Observable Results:** Verify that the DUT transmitted BREAK within 1 ms of transmitting CLOSE (NORMAL) to the Testing Station.

**Possible Problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.2.20 - CONNECTIONS THROUGH EXPANDER – OPEN SSP INITIATOR**

**Purpose:** To determine that the DUT, an SSP Initiator, can properly open a connection through an expander to transmit data or command frames.

**References:**

[1] 6.13.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** A connection is opened between a SAS initiator port and a SAS target port before communication begins. A connection is established between one SAS initiator phy in the SAS initiator port and one SAS target phy in the SAS target port. SSP initiator ports open SSP connections to transmit SCSI commands, task management functions, or transfer data. SMP initiator ports open SMP connections to transmit SMP requests and receive SMP responses. STP initiator ports and STP target ports open STP connections to transmit SATA frames.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Edge Expander.
2. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a REPORT\_GENERAL SMP Request. The Testing Station is instructed to respond on the same connection with a REPORT\_GENERAL SMP Response indicating that there are 2 attached phys. One of these phys will simulate an attached SSP target. The other phy will represent the phy that the DUT is attached to. When the response is complete close the connection.
3. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a DISCOVER SMP Request. The Testing Station is instructed to respond on the same connection with a DISCOVER SMP Response providing a SAS address of one attached SSP phy. Close the connection.
4. Wait for the DUT to open an SSP connection to the Testing Station to the same address that was provided by the Testing Station in the DISCOVER
5. Before transmitting OPEN\_ACCEPT the Testing Station is instructed to transmit AIP(NORMAL) followed by AIP(WAITING ON DEVICE)
6. Allow the DUT to transmit a SCSI Command frame to the Testing Station.
7. When the DUT transmits CLOSE (NORMAL) to close the connection, the Testing Station is instructed to transmit CLOSE (NORMAL) in response.

**Observable Results:** Verify that the DUT successfully transmitted the SCSI Command frame.

**Possible Problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.2.21 - CONNECTIONS THROUGH EXPANDERS – OPEN SMP INITIATOR**

**Purpose:** To determine that the DUT, an SMP Initiator, can properly open a connection through an expander to transmit an SMP Request.

**References:**

[1] 6.13.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** A connection is opened between a SAS initiator port and a SAS target port before communication begins. A connection is established between one SAS initiator phy in the SAS initiator port and one SAS target phy in the SAS target port. SSP initiator ports open SSP connections to transmit SCSI commands, task management functions, or transfer data. SMP initiator ports open SMP connections to transmit SMP requests and receive SMP responses. STP initiator ports and STP target ports open STP connections to transmit SATA frames.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Fanout Expander.
2. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a REPORT\_GENERAL SMP Request. The Testing Station is instructed to respond on the same connection with a REPORT\_GENERAL SMP Response indicating that there is 1 attached phy. Close the connection.
3. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a DISCOVER SMP Request. The Testing Station is instructed to respond on the same connection with a DISCOVER SMP Response providing a SAS address of one attached SMP phy. Close the connection.
4. Wait for the DUT to open an SMP connection to the Testing Station to the same address that was provided by the Testing Station in the DISCOVER
5. Before transmitting OPEN\_ACCEPT the DUT should transmit AIP(NORMAL) followed by AIP(WAITING ON DEVICE)
6. Allow the DUT to transmit an SMP Request to the Testing Station.
7. When the DUT transmits CLOSE (NORMAL) to close the connection, the Testing Station is instructed to transmit CLOSE (NORMAL) in response.

**Observable Results:** Verify that the DUT successfully transmitted the SMP Request frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.22 - CLOSING SSP CONNECTIONS – CLOSE (CLEAR AFFILIATION)**

**Purpose:** To determine that the DUT, an SSP Initiator transmits BREAK after the OpenTimeout Timer expires.

**References:**

- [1] 6.13.2.1 SAS SPL
- [2] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** To make a connection request, the source port shall transmit an OPEN address frame through an available phy. The source phy shall transmit idle dwords after the OPEN address frame until it receives a response or aborts the connection request with BREAK. After transmitting an OPEN address frame, the source phy shall initialize and start a 1 ms Open Timeout timer. If the Open Timeout timer expires before a connection response is received, the source phy may assume the destination port does not exist and shall transmit BREAK to abort the connection request [2].

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Edge Expander.
2. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a REPORT\_GENERAL SMP Request. The Testing Station is instructed to respond on the same connection with a REPORT\_GENERAL SMP Response indicating that there are 2 attached phys. One of these phys will simulate an attached SSP target. The other phy will represent the phy that the DUT is attached to. When the response is complete close the connection.
3. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a DISCOVER SMP Request. The Testing Station is instructed to respond on the same connection with a DISCOVER SMP Response providing a SAS address of one attached SSP phy. Close the connection.
4. Wait for the DUT to open an SSP connection to the Testing Station to the same address that was provided by the Testing Station in the DISCOVER
5. The Testing Station is instructed to wait for the DUT to transmit an OpenAddress frame for an SSP connection. The Testing Station is instructed to not respond to the OpenAddress frame with OPEN\_ACCEPT or OPEN\_REJECT.

**Observable Results:** Verify that 1ms after transmitting the OpenAddress frame, the DUT transmitted BREAK.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.23 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 0)**

**Purpose:** To determine that the DUT, an SMP Initiator transmits BREAK after the OpenTimeout Timer expires.

**References:**

- [1] 6.13.2.1 SAS SPL
- [2] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** BREAK may be used to abort a connection request. The source phy shall transmit a BREAK after the Open Timeout timer expires or if it chooses to abort its request for any other reason. After transmitting BREAK, the source phy shall initialize a Break Timeout timer to 1 ms and start the Break Timeout timer. When a phy sourcing a BREAK is attached to an expander device, the BREAK response to the source phy is generated by the expander phy to which the source phy is attached, not the other SAS phy in the connection. If the expander device has transmitted a connection request to the destination, it shall also transmit BREAK to the destination.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an initiator the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Fanout Expander.
2. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a REPORT\_GENERAL SMP Request. The Testing Station is instructed to respond on the same connection with a REPORT\_GENERAL SMP Response indicating that there is 1 attached phy. Close the connection.
3. Wait for the DUT to open an SMP Connection to the Testing Station. Allow the DUT to transmit a DISCOVER SMP Request. The Testing Station is instructed to respond on the same connection with a DISCOVER SMP Response providing a SAS address of one attached SMP phy. Close the connection.
4. Wait for the DUT to open an SMP connection to the Testing Station to the same address that was provided by the Testing Station in the DISCOVER
5. The Testing Station is instructed to wait for the DUT to transmit an OpenAddress frame for an SMP connection. The Testing Station is instructed to not respond to the OpenAddress frame with OPEN\_ACCEPT or OPEN\_REJECT.

**Observable Results:** Verify that 1 ms after transmitting the OpenAddress frame, the DUT transmitted BREAK.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.2.24 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 1)**

**Purpose** – To determine that the DUT properly responds when CLOSE (RESERVED 1) is received.

**Reference:**

- [1] 6.2.6.5 SAS SPL
- [2] 6.13.6 SAS SPL
- [3] 6.17.8.1 SAS SPL
- [4] 6.18.6 SAS SPL
- [5] 6.19.5 SAS SPL
- [6] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [3] for details on closing SSP connections, [4] for details on closing STP connections, and [5] for details on closing SMP connections. After transmitting CLOSE (RESERVED 1), the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

CLOSE (RESERVED 0) closes an open STP connection and clears the affiliation [1][6]. CLOSE (RESERVED 1) is processed the same as CLOSE (NORMAL) if:

- a) the connection is not an STP connection;
- b) the connection is an STP connection, but affiliations are not implemented by the STP target port; or
- c) the connection is an STP connection, but an affiliation is not present.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (RESERVED 1) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (RESERVED 1) from the Testing Station.

**Possible Problems:** None

### **GROUP 3: LINK LAYER TESTS FOR EXPANDERS**

**Overview:**

This group of tests verifies the Link Layer specifications of the SAS physical layer defined in Clause 6 of the SAS SPL, for Expanders.

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.1 - LINK RESET – REPEAT PHY SEQUENCE IF NOT IDENTIFY RECEIVED**

**Purpose:** To determine that the DUT will repeat the phy reset sequence if an Identify frame is not received within 1 ms of completing a phy reset sequence.

**References:**

- [1] 6.10.1 SAS SPL
- [2] 6.8.2 SAS SPL
- [3] 4.4.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** After the phy reset sequence has been completed indicating the physical link is using SAS rather than SATA, each phy transmits either: a) an IDENTIFY address frame [2]; or b) a HARD\_RESET. Each phy receives an IDENTIFY address frame or a HARD\_RESET from the phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence, and an identification sequence is called a link reset sequence [3]. If a device does not receive a valid IDENTIFY address frame within 1 ms of phy reset sequence completion, it shall restart the phy reset sequence.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit valid SAS primitives or SAS Idle. The Testing Station is instructed to not transmit an Identify Address Frame.

**Observable Results:**

- Verify that 1 ms after the DUT completed a phy Reset Sequence with the DUT, the DUT initiated another phy reset sequence by transmitting COMINIT.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.2 - IGNORE ADDITIONAL IDENTIFY RECEIVED**

**Purpose:** To determine that the DUT will ignore an Identify frame received after a valid Identify has already been received.

**References:**

- [1] 6.10.1 SAS SPL
- [2] 6.8.2 SAS SPL
- [3] 4.4.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** After the phy reset sequence has been completed indicating the physical link is using SAS rather than SATA, each phy transmits either: a) an IDENTIFY address frame [2]; or b) a HARD\_RESET. Each phy receives an IDENTIFY address frame or a HARD\_RESET from the phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence, and an identification sequence is called a link reset sequence [2]. If a device receives an additional IDENTIFY address frame after receiving the first one, without an intervening phy reset sequence, it shall ignore the additional IDENTIFY address frame.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit valid SAS primitives or SAS Idle. The Testing Station is instructed to transmit 2 valid Identify Address Frames each with a different SAS address.

**Observable Results:**

- Verify that the DUT ignores the second received Identify Address Frame and does not transmit COMINIT.
- Verify that the DUT does not attempt to open a connection to the address in the second Identify Address frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.3 - LINK RESET – HARD\_RESET RECEIVED**

**Purpose:** To determine that the DUT properly handles a received HARD\_RESET.

**References:**

- [1] 6.2.5.3 SAS SPL
- [2] 6.10.1 SAS SPL
- [3] 4.4 SAS SPL
- [4] 4.4.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** HARD\_RESET is used to force a phy to generate a hard reset to its port. This primitive is only valid after the phy reset sequence without an intervening identification sequence [3] and shall be ignored at other times. If a phy receives a HARD\_RESET, it shall be considered a reset event and cause a hard reset [4] of the port containing that phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit 6 consecutive HARD\_RESET primitives. This must occur before the Identify sequence is complete.

**Observable Results:**

- Verify that the DUT transmits COMINIT upon receiving the HARD\_RESET primitives from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.4 - CONNECTIONS – OPEN\_ACCEPT**

**Purpose:** To determine that the DUT properly transmits RRDY after receiving OPEN\_ACCEPT.

**References:**

[1] 6.19.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** The OPEN address frame is used to request that a connection be opened. AIP, OPEN\_ACCEPT and OPEN\_REJECT are the responses to an OPEN address frame. An OPEN\_ACCEPT indicates that the connection request was accepted. This is sent by the destination phy.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander the Testing Station is instructed to transmit an Identify Address frame indicating that it is an SMP initiator.
2. The Testing Station is instructed to transmit an SMP Open Address frame to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT.

**Observable Results:**

- Verify that the DUT OPEN\_ACCEPT within 1 ms of receiving the OPEN address frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.5 - CONNECTIONS – 2 SOAF RECEIVED**

**Purpose:** To determine that the DUT properly handles received 2 SOAF primitives.

**References:**

[1] 6.19 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** A complete OPEN ADDRESS frame will start with an SOAF primitive and end with an EOAF primitive. A SAS device must be capable of detecting an OPEN ADDRESS frame in a stream of SAS primitives.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander the Testing Station is instructed to transmit an Identify Address frame indicating that it is an SMP Initiator.
2. The Testing Station is instructed to transmit the following to the DUT
  - a. SOAF primitive
  - b. the first 4 dwords of an SMP OpenAddress frame
  - c. complete SMP Open Address frame starting with SOAF.
3. Wait for the DUT to transmit OPEN\_ACCEPT.

**Observable Results:**

- Verify that the DUT responded to the received, complete Open Address frame with OPEN\_ACCEPT.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.6 - CONNECTIONS - OPEN\_REJECT NO DESTINATION**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

- [1] 6.8 SAS SPL
- [2] 8.4.3.12 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. . OPEN\_REJECT (NO DESTINATION) occurs when:

- No such destination device;
- the expander device determines the connection request would have to be routed to the same expander port as the expander port through which the connection request arrived (e.g., the destination SAS address equals the source SAS address) and the expander device has not chosen to return OPEN\_REJECT (BAD DESTINATION) or
- the SAS address is valid for an STP target port in an STP/SATA bridge, but the initial Register - Device to Host FIS has not been successfully received .

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to SAS Expanders.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to transmit an SMP OpenAddress frame to the DUT with an invalid destination address.
3. Wait for the DUT to transmit OPEN\_REJECT (No Destination)

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (No Destination).

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.7 - CONNECTIONS - OPEN\_REJECT BAD DESTINATION**

**Purpose:** To determine that the DUT handles errors in OpenAddress frames properly.

**References:**

[1] 6.8 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (BAD DESTINATION) occurs when the destination SAS address does not match the SAS address of the SAS port to which the connection request was delivered.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to SAS Expanders.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander the Testing Station is instructed to transmit an Identify Address frame indicating that it is an SMP initiator.
2. The Testing Station is instructed to transmit an SMP OpenAddress frame to the DUT with the destination SAS address equal to the source SAS Address
3. Wait for the DUT to transmit OPEN\_REJECT (Bad Destination)

**Observable Results:** Verify that the DUT responded to the received, complete Open Address frame with OPEN\_REJECT (Bad Destination).

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.8 - CONNECTIONS – OPEN STP TARGET**

**Purpose:** To determine that the DUT, an STP Target, can properly open a connection through an expander to transmit frames.

**References:**

[1] 6.13.1 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** A connection is opened between a SAS initiator port and a SAS target port before communication begins. A connection is established between one SAS initiator phy in the SAS initiator port and one SAS target phy in the SAS target port. For STP connections, connections may be between the STP initiator port and an STP target port in an expander device attached to a SATA device. An STP target port in an expander device opens STP connections on behalf of SATA devices.

**Test Setup:** The DUT and the Testing Station are physically connected. A SATA Target device is attached to the DUT.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence and an Identify sequence with the DUT. The Testing Station is instructed to indicate that it is an Edge Expander in the Identify sequence.
2. The Testing Station is instructed to simulate an SMP Initiator attached to the Edge Expander and open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The Testing Station is instructed to transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SATA Device Port. The Testing Station is instructed to transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to open an STP Connection to the DUT for the attached SATA Device Port and transmit an IDENTIFY DEVICE command, then close the connection.
5. The Testing Station is instructed to wait for the DUT to transmit an OpenAddress frame for an STP connection. The Testing Station is instructed to respond to the OpenAddress frame with OPEN\_ACCEPT.

**Observable Results:** Verify that the DUT successfully transmitted the response to the IDENTIFY DEVICE command and completed with status OK.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.9 - CONNECTIONS – OPEN REJECT (STP RESOURCES BUSY)**

**Purpose:** To determine that the DUT, an expander acting as an STP Target properly uses OPEN REJECT (STP RESOURCES BUSY).

**References:**

- [1] 6.8 SAS SPL
- [2] 6.18.4 SAS SPL
- [3] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** OPEN\_REJECT specifies that a connection request has been rejected and specifies the reason for the rejection. The result of some OPEN\_REJECTs is to abandon (i.e., not retry) the connection request and the result of other OPEN\_REJECTs is to retry the connection request. OPEN\_REJECT (STP RESOURCES BUSY) occurs when a STP target port with destination SAS address exists but the STP target port has an affiliation with another STP initiator port or all of the available task file registers have been allocated to other STP initiator ports [3]. Process the same as OPEN\_REJECT (WRONG DESTINATION) for non-STP connection requests.

**Test Setup:** The DUT and the Testing Station are physically connected. A SATA disk drive is attached to the DUT.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached SATA device the Testing Station is instructed to transmit an Identify Address frame indicating that it is a SAS Initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT. The Testing Station is instructed to transmit a REPORT\_GENERAL SMP Request to the DUT. The DUT should respond on the same connection with a REPORT\_GENERAL SMP Response. Close the connection.
3. The Testing Station is instructed to open an SMP Connection to the DUT. The Testing Station is instructed to transmit a DISCOVER SMP Request to the DUT with the phy ID of the expander phy with the attached SATA device. The DUT should respond on the same connection with a DISCOVER SMP Response. Close the connection.
4. The Testing Station is instructed to transmit an STP OPEN ADDRESS frame to the DUT. The Testing Station is instructed to wait for OPEN\_ACCEPT from the DUT.
5. The Testing Station is instructed to transmit a second STP OPEN ADDRESS frame to the DUT, from the same SAS source address.

**Observable Results:** Verify that the DUT transmits OPEN REJECT (STP RESOURCES BUSY) in response to the second received OPEN ADDRESS frame. If the DUT tracks all commands by the STP initiator ports' SAS addresses, this item is not applicable. This item is only applicable if the DUT supports affiliations.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.10 - CONNECTIONS – SATA\_HOLD**

**Purpose:** To determine that the DUT, an expander acting as an SATA Host properly uses SATA\_HOLD.

**References:**

[1] 6.18.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** Each STP port (i.e., STP initiator port and STP target port) and expander port through which the STP connection is routed shall implement the SATA flow control protocol on each physical link. The flow control primitives are not forwarded through expander devices like other dwords. When an STP port is receiving a frame and its buffer begins to fill up, it shall transmit SATA\_HOLD. When an STP port is transmitting a frame and receives SATA\_HOLD, it shall transmit no more than 20 data dwords for the frame and respond with SATA\_HOLDA.

**Test Setup:** The DUT and the Testing Station are physically connected. A SATA disk drive is attached to the DUT.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached SATA device the Testing Station is instructed to transmit an Identify Address frame indicating that it is a SAS Initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT. The Testing Station is instructed to transmit a REPORT\_GENERAL SMP Request to the DUT. The DUT should respond on the same connection with a REPORT\_GENERAL SMP Response. Close the connection.
3. The Testing Station is instructed to open an SMP Connection to the DUT. The Testing Station is instructed to transmit a DISCOVER SMP Request to the DUT with the phy ID of the expander phy with the attached SATA device. The DUT should respond on the same connection with a DISCOVER SMP Response. Close the connection.
4. The Testing Station is instructed to transmit an STP OPEN ADDRESS frame to the DUT. The Testing Station is instructed to wait for OPEN\_ACCEPT from the DUT.
5. The Testing Station is instructed to begin transmission of a SATA FIS of IDENTIFY DEVICE, but interrupt transmission of that FIS with SATA\_HOLD primitives.
6. The Testing Station is instructed to wait for the DUT to transmit SATA\_HOLDA, then complete transmission of the SATA FIS.

**Observable Results:** Verify that the DUT retransmits SATA\_HOLD and the SATA\_HOLDA

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.11 - CONNECTIONS – SATA\_RIP**

**Purpose:** To determine that the DUT, an expander acting as an SATA Host properly uses SATA\_RIP.

**References:**

[1] 6.18.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** The SATA\_RIP primitive allows STP devices to indicate that frame reception is occurring. A device will source SATA\_RIP after a device sources SATA\_RRDY, indicating it is ready to receive a frame. SATA\_RIP is an STP primitive which allows SAS devices to communicate with SATA devices.

**Test Setup:** The DUT and the Testing Station are physically connected. A SATA disk drive is attached to the DUT.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached SATA device the Testing Station is instructed to transmit an Identify Address frame indicating that it is a SAS Initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT. The Testing Station is instructed to transmit a REPORT\_GENERAL SMP Request to the DUT. The DUT should respond on the same connection with a REPORT\_GENERAL SMP Response. Close the connection.
3. The Testing Station is instructed to open an SMP Connection to the DUT. The Testing Station is instructed to transmit a DISCOVER SMP Request to the DUT with the phy ID of the expander phy with the attached SATA device. The DUT should respond on the same connection with a DISCOVER SMP Response. Close the connection.
4. The Testing Station is instructed to transmit an STP OPEN ADDRESS frame to the DUT. The Testing Station is instructed to wait for OPEN\_ACCEPT from the DUT.
5. Wait for the DUT to transmit SATA\_R\_RDY.
6. The Testing Station is instructed to transmit SATA\_XRDY, followed by an IDENTIFY DEVICE command to the DUT. Wait for the attached SATA device to transmit SATA\_RIP.

**Observable Results:** Verify that the DUT forwards SATA\_RIP from the attached SATA device.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.12 - BREAK – OPENTIMEOUT TIMER STP TARGET**

**Purpose:** To determine that the DUT, an STP Target transmits BREAK after the OpenTimeout Timer expires.

**References:**

- [1] 6.13.2.1 SAS SPL
- [2] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator, SATA Device Port

**Last Modification:** July 26, 2011

**Discussion:** To make a connection request, the source port shall transmit an OPEN address frame through an available phy. The source phy shall transmit idle dwords after the OPEN address frame until it receives a response or aborts the connection request with BREAK. After transmitting an OPEN address frame, the source phy shall initialize and start a 1 ms Open Timeout timer. If the Open Timeout timer expires before a connection response is received, the source phy may assume the destination port does not exist and shall transmit BREAK to abort the connection request [2].

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is also connected to a SATA Device Port.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence and an Identify sequence with the DUT. The Testing Station is instructed to indicate that it is an Edge Expander in the Identify sequence.
2. The Testing Station is instructed to simulate an SMP Initiator attached to the Edge Expander and open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The Testing Station is instructed to transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SATA Device Port. The Testing Station is instructed to transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to open an STP Connection to the DUT for the attached SATA Device Port and transmit an IDENTIFY DEVICE command, then close the connection.
5. The Testing Station is instructed to wait for the DUT to transmit an OpenAddress frame for an STP connection. The Testing Station is instructed to not respond to the OpenAddress frame with OPEN\_ACCEPT or OPEN\_REJECT.

**Observable Results:** Verify that 1ms after transmitting the OpenAddress frame, the DUT transmitted BREAK.

**Possible Problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.3.13 - BREAK – CONNECTION REQUEST NOT TRANSMITTED**

**Purpose:** To determine that the DUT, a SAS Expander responds to received BREAK properly.

**References:**

- [1] 6.13.2.1 SAS SPL
- [2] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** When a phy sourcing a BREAK is attached to an expander device, the BREAK response to the source phy is generated by the expander phy to which the source phy is attached, not the other SAS phy in the connection. If the expander device has transmitted a connection request to the destination, it shall also transmit BREAK to the destination. If the expander device has not transmitted a connection request to the destination, it shall not transmit BREAK to the destination. After transmitting BREAK back to the originating phy, the expander device shall ensure that an open response does not occur (i.e., the expander device shall not forward dwords from the destination any more).

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, followed by a BREAK, within 10 dwords.

**Observable Results:** Verify that the DUT does not transmit BREAK to the SAS Target unless it has already transmitted the SSP OpenAddress frame to the target. Verify that after transmitting BREAK back to the Testing Station, the DUT does not forward any dwords from the attached SAS Target to the Testing Station, such as an OPEN\_ACCEPT frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.14 - BREAK – CONNECTION REQUEST TRANSMITTED**

**Purpose:** To determine that the DUT, a SAS Expander responds to received BREAK properly.

**References:**

- [1] 6.13.2.1 SAS SPL
- [2] 6.13.6 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** When a phy sourcing a BREAK is attached to an expander device, the BREAK response to the source phy is generated by the expander phy to which the source phy is attached, not the other SAS phy in the connection. If the expander device has transmitted a connection request to the destination, it shall also transmit BREAK to the destination. If the expander device has not transmitted a connection request to the destination, it shall not transmit BREAK to the destination. After transmitting BREAK back to the originating phy, the expander device shall ensure that an open response does not occur (i.e., the expander device shall not forward dwords from the destination any more).

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT. Once OPEN\_ACCEPT is received from the DUT, the Testing Station is instructed to transmit BREAK.

**Observable Results:** Verify that the DUT transmits BREAK to the SAS Target.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.15 - AIP –TRANSMITTED**

**Purpose:** To determine that the DUT, a SAS Expander properly transmits AIP

**References:**

- [1] 6.13.2.2 SAS SPL
- [2] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** When an expander device is trying to open a connection to the selected destination port, it returns an AIP to the source phy. The source phy shall reinitialize and restart its Open Timeout timer when it receives an AIP. AIP is sent by an expander device while it is internally arbitrating for access to an expander port. Expander devices shall transmit no more than three consecutive AIPs without transmitting an idle dword. Expander devices may transmit three consecutive AIPs to provide better tolerance of errors. Expander devices shall transmit at least one AIP every 128 dwords while transmitting AIP (NORMAL), AIP (WAITING ON PARTIAL), or AIP (WAITING ON CONNECTION).

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT.

**Observable Results:** Verify that the DUT transmits AIP to the Testing Station upon receiving the connection request. The AIP must be transmitted within 128 dwords of the Open, and AIP must be sent at least every 128 dwords until the connection is established from the DUT to the attached SAS Target.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.16 - AIP –OPEN RECEIVED HIGHER PRIORITY**

**Purpose:** To determine that the DUT, a SAS Expander properly handles a received OPEN of higher priority after having transmitted AIP on a different physical link.

**References:**

- [1] 6.13.2.2 SAS SPL
- [2] 6.18.4 SAS SPL
- [3] 6.18.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** When an expander device is trying to open a connection to the selected destination port, it returns an AIP to the source phy. The source phy shall reinitialize and restart its Open Timeout timer when it receives an AIP. AIP is sent by an expander device while it is internally arbitrating for access to an expander port. Before an expander device transmits AIP, it may have transmitted an OPEN address frame on the same physical link. Arbitration fairness dictates which OPEN address frame wins [1]. After an expander device transmits an AIP, it shall not transmit an OPEN address frame unless it has higher arbitration priority than the incoming connection request.

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, and transmit a SCSI INQUIRY command. The Testing Station is instructed to close the connection.
5. The Testing Station is instructed to wait for an OpenAddress frame from the attached target. At this point the DUT must be transmitting AIP to the attached target. The Testing Station is instructed to transmit an OpenAddress to the attached target with a high priority, by setting AWT to 8000h.

**Observable Results:** Verify that the DUT transmits AIP to the attached target upon receiving the OpenAddress from the attached target. Verify that the DUT forwards the OpenAddress from the Testing Station to the attached target after having transmitted AIP to the attached target.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.17 - AIP –OPEN RECEIVED LOWER PRIORITY**

**Purpose:** To determine that the DUT, a SAS Expander properly handles a received OPEN of higher priority after having transmitted AIP on a different physical link.

**References:**

- [1] 6.13.2.2 SAS SPL
- [2] 6.18.5 SAS SPL
- [3] 6.18.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** When an expander device is trying to open a connection to the selected destination port, it returns an AIP to the source phy. The source phy shall reinitialize and restart its Open Timeout timer when it receives an AIP. AIP is sent by an expander device while it is internally arbitrating for access to an expander port. Before an expander device transmits AIP, it may have transmitted an OPEN address frame on the same physical link. Arbitration fairness dictates which OPEN address frame wins [3]. After an expander device transmits an AIP, it shall not transmit an OPEN address frame unless it has higher arbitration priority than the incoming connection request.

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, and transmit a SCSI INQUIRY command. The Testing Station is instructed to close the connection.
5. The Testing Station is instructed to wait for an OpenAddress frame from the attached target. At this point the DUT must be transmitting AIP to the attached target. The Testing Station is instructed to transmit an OpenAddress to the attached target with a low priority, by setting AWT to 0000h.

**Observable Results:** Verify that the DUT transmits AIP to the attached target upon receiving the OpenAddress from the attached target. Verify that the DUT does not forward the OpenAddress from the Testing Station to the attached target after having transmitted AIP to the attached target.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.18 - AWT TIMER –ARBITRATION WAIT TIME**

**Purpose:** To determine that the DUT, a SAS Expander, properly sets the AWT in an outgoing OpenAddress frame.

**References:**

[1] 6.18.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** Each SAS port and expander port shall include an Arbitration Wait Time timer which counts the time from the moment when the port makes a connection request until the request is accepted or rejected. The expander port that receives an OPEN address frame shall set the Arbitration Wait Time timer to the value of the incoming ARBITRATION WAIT TIME field and start the Arbitration Wait Time timer as it arbitrates for internal access to the outgoing expander port. When the expander device transmits the OPEN address frame out another expander port, it shall set the outgoing ARBITRATION WAIT TIME field to the current value of the Arbitration Wait Time timer maintained by the incoming expander port.

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, with an AWT of 5FFFh.

**Observable Results:** Verify that the DUT transmits an OpenAddress frame for an SSP Connection to the attached SAS Target with an AWT of 5FFFh.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.19 - AWT TIMER – RESET TO ZERO**

**Purpose:** To determine that the DUT, a SAS Expander, properly sets the AWT in an outgoing OpenAddress frame.

**References:**

[1] 6.18.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** Each SAS port and expander port shall include an Arbitration Wait Time timer which counts the time from the moment when the port makes a connection request until the request is accepted or rejected. A SAS port shall stop the Arbitration Wait Time timer and set it to zero when it receives one of the following connection responses:

- a) OPEN\_ACCEPT;
- b) OPEN\_REJECT (PROTOCOL NOT SUPPORTED);
- c) OPEN\_REJECT (STP RESOURCES BUSY);
- d) OPEN\_REJECT (WRONG DESTINATION); or
- e) OPEN\_REJECT (RETRY).

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, with an AWT of 5FFFh. The Testing Station is instructed to transmit a SCSI INQUIRY command to that attached target then close the connection.
5. Allow the attached target to open SSP connections through the DUT, to transmit Data and Response to the Testing Station.
6. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, with an AWT of 0.

**Observable Results:** Verify that the DUT transmits an OpenAddress frame for an SSP Connection to the attached SAS Target with an AWT of 0.

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.3.20 - AWT TIMER – ARBITRATION FAIRNESS**

**Purpose:** To determine that the DUT, a SAS Expander, properly uses the AWT field in an OpenAddress frame to determine arbitration fairness.

**References:**

- [1] 6.18.2 SAS SPL
- [2] 6.8.3 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** Each SAS port and expander port shall include an Arbitration Wait Time timer which counts the time from the moment when the port makes a connection request until the request is accepted or rejected. SAS ports should set the Arbitration Wait Time timer to zero when they transmit the first OPEN address frame for the connection request. A SAS initiator port or SAS target port may be unfair by setting the ARBITRATION WAIT TIME field in the OPEN address frame (see 6.8.3) to a higher value than its Arbitration Wait Time timer indicates. However, unfair SAS ports shall not set the ARBITRATION WAIT TIME field to a value greater than or equal to 8000h; this limits the amount of unfairness and helps prevent livelocks.

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, with an AWT of 0FFFh. Immediately following, the Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, with an AWT of 5FFFh.

**Observable Results:** Verify that the DUT transmits an OpenAddress frame for an SSP Connection to the attached SAS Target with an AWT of 5FFFh.

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.21 - OPEN\_REJECT – PATHWAY BLOCKED**

**Purpose:** To determine that the DUT, a SAS Expander, properly increments the pathway blocked count.

**References:**

[1] 6.18.4 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** Pathway recovery provides a means to abort connection requests in order to prevent deadlock using pathway recovery priority comparisons. The ECM shall instruct the arbitrating expander phy to reject the connection request by transmitting OPEN\_REJECT (PATHWAY BLOCKED) when:

- a) the Partial Pathway Timeout timer expires; and
- b) the pathway recovery priority of the arbitrating expander phy (i.e., the expander phy requesting the connection) is less than or equal to the pathway recovery priority of any of the expander phys within the destination port that are sending phy Status (Blocked Partial Pathway) responses to the ECM.

**Test Setup:** The DUT and the Testing Station are physically connected. The DUT is physically connected to a SAS Target.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a DISCOVER SMP Request to the DUT phy which is attached to the SAS Target. The DUT should transmit a DISCOVER SMP Response with this information.
4. The Testing Station is instructed to transmit an OpenAddress frame for an SSP Connection to the SAS Target attached to the DUT, transmit a SCSI INQUIRY command, then close the connection.
5. The target attached to the DUT is expected to transmit a connection request to the Testing Station through the DUT. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP, then close the connection. Response. The SMP OPEN from the Testing Station is instructed to have a higher AWT than the OPEN from the attached target. Repeat this step 5 times.

**Observable Results:** Verify that the DUT transmits OPEN\_REJECT (PATHWAY BLOCKED) to the attached target OPEN request. Verify that the DUT increments the PATHWAY BLOCKED count each time an OPEN request from the target is rejected.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.22 - SMP\_REQUEST – PROPER RESPONSE FORMAT**

**Purpose:** To determine that the DUT, a SAS Expander, properly formats an SMP Response.

**References:**

- [1] 6.18.4 SAS SPL
- [2] 8.4 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** Inside an SMP connection, the source device transmits a single SMP\_REQUEST frame and the destination device responds with a single SMP\_RESPONSE frame [2]. By accepting an SMP connection, the destination device indicates it is ready to receive one SMP\_REQUEST frame. After the source device transmits one SMP\_REQUEST frame, it shall be ready to receive one SMP\_RESPONSE frame.

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an SMP Initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.

**Observable Results:** Verify that the DUT transmits an SMP Response, which is formatted properly and it not waiting on an interlocked frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.23 - SMP\_REQUEST – CLOSE**

**Purpose:** To determine that the DUT, a SAS Expander, properly closes an SMP Connection.

**References:**

- [1] 6.19.5 SAS SPL
- [2] 8.4 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** Inside an SMP connection, the source device transmits a single SMP\_REQUEST frame and the destination device responds with a single SMP\_RESPONSE frame [2]. After receiving the SMP\_RESPONSE frame, the source device shall transmit a CLOSE (NORMAL) to close the connection. After transmitting the SMP\_RESPONSE frame, the destination device shall reply with a CLOSE (NORMAL).

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an SMP Initiator.
2. The Testing Station is instructed to open an SMP Connection to the DUT and transmit a REPORT\_GENERAL SMP Request. The DUT should transmit a REPORT\_GENERAL SMP Response.
3. The Testing Station is instructed to transmit CLOSE.

**Observable Results:** Verify that the DUT transmits a CLOSE(NORMAL) after receiving CLOSE from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire*  
*InterOperability Laboratory*

**Test 6.3.24 - SMP\_REQUEST – PROPER REQUEST FORMAT**

**Purpose:** To determine that the DUT, a SAS Expander, properly formats an SMP Request.

**References:**

- [1] 6.19.5 SAS SPL
- [2] 8.4 SAS SPL
- [3] 6.5 SAS SPL
- [4] 6.19.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator. SAS Target.

**Last Modification:** July 26, 2011

**Discussion:** Inside an SMP connection, the source device transmits a single SMP\_REQUEST frame and the destination device responds with a single SMP\_RESPONSE frame [2]. Frames are surrounded by SOF and EOF. See 6.19.5 for error handling details. The last data dword after the SOF prior to the EOF always contains a CRC [3]. The SMP link layer state machine checks that the frame is not too short and that the CRC is valid [4].

**Test Setup:** The DUT and the Testing Station are physically connected.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is an expander with an attached target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an Edge Expander, with an SMP Target.
2. The DUT should open an SMP Connection to the Testing Station and transmit a REPORT\_GENERAL SMP Request. The Testing Station is instructed to transmit a REPORT\_GENERAL SMP Response.

**Observable Results:** Verify if the DUT transmits an SMP Request, which is formatted properly and is not waiting on an interlocked frame.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.25 - CLOSING SSP CONNECTIONS – CLOSE (CLEAR AFFILIATION)**

**Purpose** – To determine that the DUT properly responds when CLOSE (CLEAR AFFILIATION) is received.

**Reference:**

- [1] 6.2.6.5 SAS SPL
- [2] 6.13.6 SAS SPL
- [3] 6.17.8.1 SAS SPL
- [4] 6.18.6 SAS SPL
- [5] 6.19.5 SAS SPL
- [6] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [3] for details on closing SSP connections, [4] for details on closing STP connections, and [5] for details on closing SMP connections. After transmitting CLOSE (CLEAR AFFILIATION), the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

CLOSE (CLEAR AFFILIATION) closes an open STP connection and clears the affiliation [1][6]. CLOSE (CLEAR AFFILIATION) is processed the same as CLOSE (NORMAL) if:

- a) the connection is not an STP connection;
- b) the connection is an STP connection, but affiliations are not implemented by the STP target port; or
- c) the connection is an STP connection, but an affiliation is not present.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (CLEAR AFFILIATION) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (CLEAR AFFILIATION) from the Testing Station.

**Possible Problems:** None



*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.26 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 0)**

**Purpose** – To determine that the DUT properly responds when CLOSE (RESERVED 0) is received.

**Reference:**

- [1] 6.2.6.5 SAS SPL
- [2] 6.13.6 SAS SPL
- [3] 6.17.8.1 SAS SPL
- [4] 6.18.6 SAS SPL
- [5] 6.19.5 SAS SPL
- [6] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [3] for details on closing SSP connections, [4] for details on closing STP connections, and [5] for details on closing SMP connections. After transmitting CLOSE (RESERVED 0), the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

CLOSE (RESERVED 0) closes an open STP connection and clears the affiliation [1][6]. CLOSE (RESERVED 0) is processed the same as CLOSE (NORMAL) if:

- a) the connection is not an STP connection;
- b) the connection is an STP connection, but affiliations are not implemented by the STP target port; or
- c) the connection is an STP connection, but an affiliation is not present.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (RESERVED 0) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (RESERVED 0) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.27 - CLOSING SSP CONNECTIONS - CLOSE (RESERVED 1)**

**Purpose** – To determine that the DUT properly responds when CLOSE (RESERVED 1) is received.

**Reference:**

- [1] 6.2.6.5 SAS SPL
- [2] 6.13.6 SAS SPL
- [3] 6.17.8.1 SAS SPL
- [4] 6.18.6 SAS SPL
- [5] 6.19.5 SAS SPL
- [6] 6.18.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** CLOSE is used to close a connection of any protocol. See [3] for details on closing SSP connections, [4] for details on closing STP connections, and [5] for details on closing SMP connections. After transmitting CLOSE (RESERVED 1), the source phy shall initialize a Close Timeout timer to 1 ms and start the Close Timeout timer. Once an SSP phy has both transmitted and received DONE, it shall close the connection by transmitting CLOSE (NORMAL).

CLOSE (RESERVED 0) closes an open STP connection and clears the affiliation [1][6]. CLOSE (RESERVED 1) is processed the same as CLOSE (NORMAL) if:

- a) the connection is not an STP connection;
- b) the connection is an STP connection, but affiliations are not implemented by the STP target port; or
- c) the connection is an STP connection, but an affiliation is not present.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open an SSP connection to the DUT. Wait for the DUT to transmit OPEN\_ACCEPT. The Testing Station is instructed to transmit a CLOSE (RESERVED 1) primitive to the DUT.

**Observable Results:** Verify that the DUT transmitted CLOSE (NORMAL) within 1 ms of receiving CLOSE (RESERVED 1) from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.28 - LINK RESET – HARD\_RESET RECIVED AFTER IDENTIFY**

**Purpose:** To determine that the DUT properly handles a received HARD\_RESET.

**References:**

- [1] 6.2.5.3 SAS SPL
- [2] 6.10.1 SAS SPL
- [3] 4.4 SAS SPL
- [4] 4.4.2 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** HARD\_RESET is used to force a phy to generate a hard reset to its port. This primitive is only valid after the phy reset sequence without an intervening identification sequence [3] and shall be ignored at other times. If a phy receives a HARD\_RESET, it shall be considered a reset event and cause a hard reset [4] of the port containing that phy.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to transmit COMINT to start a phy Reset sequence with the DUT.
2. The Testing Station is instructed to complete a phy Reset sequence with the DUT.
3. Once the phy reset sequence between the DUT and the Testing Station are complete, the Testing Station is instructed to transmit 6 consecutive HARD\_RESET primitives. This must occur after the Identify sequence is complete.

**Observable Results:** Verify that the DUT does not transmit COMINIT upon receiving the HARD\_RESET primitives from the Testing Station.

**Possible Problems:** None

*The University of New Hampshire  
InterOperability Laboratory*

**Test 6.3.29 - SSP\_FRAMES – INTERLOCKED FRAME / NAK**

**Purpose:** To determine that the DUT handles interlocked frames with CRC errors properly.

**References:**

- [1] 6.17.3 SAS SPL
- [2] 6.16.5 SAS SPL
- [3] 6.2.6.5 SAS SPL

**Resource Requirements:** SAS Protocol Analyzer and Generator

**Last Modification:** July 26, 2011

**Discussion:** Before transmitting an interlocked frame, an SSP phy shall wait for all SSP frames to be acknowledged with ACK or NAK, even if credit is available. After transmitting an interlocked frame, an SSP phy shall not transmit another SSP frame until it has been acknowledged with ACK or NAK, even if credit is available. A receiving SSP phy shall respond to a SSP frame with NAK (CRC ERROR) if the SSP frame was received with a bad CRC.

**Test Setup:** The DUT and the Testing Station are physically connected. This test is only applicable to targets.

**Test Procedure:**

1. The Testing Station is instructed to start and complete a phy Reset sequence followed by an Identify sequence with the DUT. Since the DUT is a target the Testing Station is instructed to transmit an Identify Address frame indicating that it is an initiator.
2. The Testing Station is instructed to open a SSP connection to the DUT and transmit a SCSI\_INQUIRY command with an invalid CRC.
3. Wait for the DUT to transmit NAK (CRC ERROR).

**Observable Results:** Verify that upon receiving the INQUIRY command with an invalid CRC, the DUT transmitted NAK (CRC ERROR) before transmitting any other frame to the Testing Station.

**Possible Problems:** None