LSI LOGIC®

4420 ArrowsWest Drive
Colorado Springs, CO 80907

April 13, 1999

To: T10 Technical Committee
From: John Lohmeyer, LSI Logic Principal Member of T10
Subj: Proposed Integrity Checking Annex for SPI-3

The proposals received by T10 so far on integrity checking have only addressed enhancements to the READ BUFFER and WRITE BUFFER commands in the SPC-2 draft standard. SPI-3 is completely silent on this important feature. I think this omission needs to be corrected.

I have drafted an informative annex on integrity checking. Since it is informative, it contains no actual requirements. However, it does define several terms and it provides some minimal guidance on performing integrity checks. Please review this proposed annex.

Revision 1 incorporates suggestions received during the November '98 T10 meeting week plus some additional changes as a result of the UBF proposal (99-102r0).

Revision 2 incorporates suggestions received during the January '99 SPI-3 working group meeting.

Revision 3 incorporates suggestions received during the March '99 T10 meeting week.

Revision 4 incorporates suggestions received at the April '99 SPI-3 working group meeting, which recommended that T10 accept it for inclusion in SPI-3.

Physical Layer Integrity Checking

(Informative)

1. Introduction

'Integrity checking' is the act of verifying that the physical layer is able to transfer test data at the negotiated speed and width between the initiator and target — It is a quick check for physical domain validation. For example, two wide SCSI devices connected with a narrow cable will discover that the cable does not support wide transfers during this checking. These SCSI devices will then re-negotiate to narrow transfers.

'Fall back' is the act of re-negotiating to a set of physical parameters that are less demanding. After falling back, integrity checking is again performed to verify the new parameters. This cycle may be repeated until an acceptable set of physical parameters is found.

This annex defines integrity checking and fall back terminology. Tools used to perform integrity checking include:

a) the INQUIRY command,
b) enhancements to the READ BUFFER and WRITE BUFFER commands, and
c) the Unexpected Bus Free timer in the Control Mode Page.

Integrity checking is not intended to eliminate the need for good system design; it is intended to help detect invalid configurations, where feasible. The Desktop Management Task Force (DMTF) has additional initiatives in the Mass Storage Working Group related to integrity checking.[1]

2. Integrity checking methods

2.1 Basic integrity check

The basic integrity check consists of issuing an INQUIRY command to a device three times; twice with the physical parameters set to asynchronous, narrow mode and once with the physical parameters set to the highest supported values. The first 36 bytes of returned data is compared and any detected transfer errors are noted. Should the data be equal with no errors detected, then the basic integrity check passes. Should the data not compare but no detected errors occur, then the test should be repeated (this could be due to the target changing the INQUIRY data during device initialization). Otherwise, this test fails and fall back should be attempted.

This test detects most basic problems including:

1) Path width errors (i.e., narrow cable used with wide SCSI devices)
2) Expander errors (e.g., expanders not capable of the negotiated data rate)
3) Gross cable errors (e.g., broken wire)
4) Incorrect termination (e.g., missing or bad terminator)
5) Damaged transceiver.

2.2 Enhanced integrity check

The enhanced integrity check consists of sending and receiving known data patterns using the READ BUFFER and WRITE BUFFER commands, preferably with the echo buffer option.

During these tests, the application client should prevent other processes from using the target device. The application client should use the RESERVE command to prevent other initiators from altering the data buffer in the target.

Some data patterns are more stressful on the physical layer. At a minimum, it is recommended that the application client use the following data patterns:

1) Counting (0001h, 0203h, 0405h, …)
2) Alternating ones and zeros (0000h, FFFFh, 0000h, FFFFh, …)
3) Crosstalk (5555h, AAAAh, 5555h, AAAAh, …)
4) Shifting bit (0000h, FFFEh, 0000h, FFFDh, … then FFFFh, 0001h, FFFFh, 0002h, …)

This test may detect additional problems including:

1) Wrong impedance cables
2) Bad device spacing
3) Poor termination
4) Marginal transceivers
5) Excessive crosstalk
6) Excessive system noise.

2.3 Margined integrity check

A margined integrity check verifies that the negotiated physical parameters have some degree of margin. Known data patterns are transferred with slightly altered signals to verify that no errors occur on the transfers. The assumption is that if no errors occur with the altered signals, then

[1] The DMTF may be contacted at www.dmtf.org.

transfers with normal signals should have some margin to accommodate noise not present during the testing. Should errors occur with the altered signals, then the initiator should fall back to a lower transmission speed. Altered signals should only be used during the diagnostic check and not during normal operation.

This annex does not specify which techniques to use to alter signals.

Margined integrity checking is done using the techniques described in 2.2 except that the signals are altered.

Margined integrity checking may give added confidence that the domain is sufficient to support the negotiated physical parameters.

3. Fall back

Fall back is the act of re-negotiating to a less-demanding set of physical parameters for example transfer mode reduction or bus width reduction. It is accomplished by either a PPR negotiation or a WDTR/SDTR negotiation.

4. System considerations

SCSI devices that do not implement the READ BUFFER and WRITE BUFFER commands should report CHECK CONDITION status and ILLEGAL REQUEST sense key in response to attempts to issue these commands. It may be impractical to perform certain integrity checks with these SCSI devices.

4.1 Buffer protection

The READ BUFFER and WRITE BUFFER commands access physical buffers in the target. Many implementations do not protect the buffer contents if there is an intervening command from any other process. Therefore, the application client should ensure that no other SCSI processes are active while performing tests.

The RESERVE command may be used to block commands from other initiators. However, using the RESERVE command is not sufficient to prevent commands from the same initiator (possibly issued by other processes) from corrupting the buffer contents. Also, targets with multiple logical units may corrupt the buffer if commands are processed on other logical units.

The READ BUFFER and WRITE BUFFER commands include an echo buffer option that may be especially valuable when performing these tests.

4.2 Failure modes during integrity checking

Integrity checking may cause several kinds of error conditions:

a) Parity or CRC errors - detected error
b) Data comparison mismatches - undetected error
c) Bus hangs. - requires special handling

Bus hangs occur when the target fails to detect an ACK pulse from the initiator (possibly as a result of the initiator failing to detect a REQ pulse from the target). This is a frequent failure mode on marginal physical domains. It is recommended that initiators include provisions to avoid extended bus hangs. Two recovery actions are possible:

a) Assert the RST signal

b) Use the Synchronous Transfer Timeout (STT) function in the SCSI Port Control Mode page.