Working Draft

X3T10/1224-TR

Revision 4
March 7, 1997

Information Technology - Profile for Parallel SCSI Components Used in High Availability Environments

A draft ANSI Technical Report prepared by Accredited Standards Committee X3.

ABSTRACT
This technical report defines a profile for the use of parallel SCSI equipment in environments where high system availability is required.

Technical Editor:
Douglas Hagerman
Digital Equipment Corporation
SHR3-2/C5
334 South Street
Shrewsbury MA 01545
Voice: 508-841-2145
FAX: 508-841-6100
EMail hagerman@starch.enet.dec.com

Reference number
ISO/IEC ***** : 199x
ANSI X3 TR ** 9x
Printed 03/24/97

Other Points of Contact:

	X3T10 Chairman	X3T10 Vice-Chair
	John Lohmeyer	Lawrence Lamers
	Symbios Logic Inc	Adaptec
	4420 ArrowsWest Drive	MS 293
	Colorado Springs, CO 80907-3444	691 South Milpitas Blvd
		Milpitas CA, 95035
Voice:	719-533-7560	408-957-7817
Fax:	719-533-7036	408-957-7193
Email:	john.lohmeyer@symbios.com	ljlamers@aol.com

X3 Secretariat

X3 Secretariat
1250 Eye Street, NW Suite 200
Washington, DC 20005

Voice: 202-737-8888
FAX: 202-638-4922
Email: x3sec@itic.nw.dc.us

Reflector

Internet address for subscription to the X3T10 reflector:	majordomo@symbios.com
Mail message body should contain a line stating:	subscribe scsi
Internet address for distribution via X3T10 reflector:	scsi@symbios.com

X3T10 Bulletin Board		719-533-7950
FTP Site:		ftp.symbios.com /pub/standards/io/x3t10
Web sites:		http://www.x3.org/
	or	http://www.symbios.com/x3t10/

Document Distribution
Global Engineering
15 Inverness Way East
Englewood, CO 80112-5704

Voice: 303-792-2181
or: 800-854-7179 [put this one first]
FAX: 303-792-2192

PATENT STATEMENT

CAUTION: The developers of this technical report have requested that holders of patents that may be required for the implementation of the technical report, disclose such patents to the publisher. However, neither the developers nor the publisher have undertaken a patent search in order to identify which, if any, patents may apply to this technical report.

As of the date of publication of this technical report, following calls for the identification of patents that may be required for the implementation of the technical report, notice of one or more claims has been received.

By publication of this technical report, no position is taken with respect to the validity of this claim or of any rights in connection therewith. The known patent holder(s) has (have), however, filed a statement of willingness to grant a license under these rights on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license. Details may be obtained from the publisher.

No further patent search is conducted by the developer or the publisher in respect to any technical report it processes. No representation is made or implied that licenses are not required to avoid infringement in the use of this technical report.

Contents

Error! Bookmark not defined.

Figures

X3's Technical Report Series

As a complementary product of the standards development process and the resources of knowledge devoted to it, X3 from time to time produces technical reports. Technical reports are informational or tutorial in nature. They are not standards, nor are they intended to be used as such. They are produced, in some cases, to disseminate the technical and logical concepts reflected in standards already published or under development. In other cases, they derive from studies in areas where it is found premature to develop a standard due to a still changing technology, or inappropriate to develop a rigorous standard due to the existence of a number of viable options, the choice of which depends on the users particular requirements. These technical reports, thus, provide guidelines, the use of which can result in greater consistency and coherence of information processing systems.

Publication of this ANSI Technical Report [has been] will be approved by Accredited Standards Committee X3, Information Technology. This document [is] will be registered as a Technical Report series of publications according to the Procedures for the Registration of ANSI Technical Reports. This document is not an American National Standard and the material contained herein is not normative in nature. Comments on the content of this document should be sent to the X3 Secretariat, 1250 Eye Street, NW (Suite 200), Washington, DC 20005.

Foreword

The SCSI standard describes a method for the connection of mass storage devices to servers and workstations. SCSI is the preferred method of interconnecting high-performance disk drives, tape drives, and related devices such as CD-ROM drives. The SCSI interface provides both high performance and high functionality.

The SCSI interface is suitable for use on SCSI systems where a high degree of system availability is required. Using SCSI components in such demanding systems puts strenuous requirements on both the operational details of the SCSI components as well as the overall system design. These requirements generally may be stated in the form of "good behavior" requirements that allow the system to continue to operate even in the event of certain failures.

The benefit of SCSI is that products that are in wide use in general purpose SCSI systems may be used, with little or no additional cost or complication, in the most highly demanding computing environments.

The purpose of this technical report is to provide guidance to component, device, and system designers so that they can maximize the usefulness of SCSI in high availability systems. It includes an overall description of how SCSI may be used to create high availability systems, lists the generic goals and top-level requirements on components to be used in such systems, and provides a detailed set of implementation requirements that reflect the generic goals.

This technical report was developed by Task Group X3T10 of Accredited Standards Committee X3 during 1996-97.

Requests for interpretation, suggestions for improvement and addenda, or defect reports are welcome. They should be sent to the X3 Secretariat, Information Technology Industry Council, 1250 Eye Street, NW, Suite 200, Washington, DC 20005-3922.

This technical report [was] will be processed and approved for submittal to ANSI by Accredited Standards Committee on Information Processing Systems, X3. Committee approval of the technical report does not necessarily imply that all committee members voted for approval. At the time it approved this technical report, the X3 Committee had the following members:

James D. Converse, Chair

Donald C. Loughry, Vice-Chair

Joanne M. Flanagan, Secretary

Technical Committee X3T10 on I/O Interfaces, which reviewed this technical report, had the following members:

John B. Lohmeyer, Chair Lawrence J. Lamers, Vice-Chair Ralph Weber, Secretary

[names to be added]

Introduction

This technical report is divided into the following clauses and annexes.

Clause 1 defines the scope of the document.

Clause 2 explains the relationship between technical reports and Standards.

Clause 3 defines the definitions, keywords, and conventions.

Clause 4 defines objects and object notation.

Clause 5 defines High Availability SCSI Systems.

Clause 6 defines Multi-Host SCSI Systems.

Clause 7 defines the Fundamental Requirements for high availability multi-host SCSI Systems.

Clause 8 defines the System Level Requirements.

Clause 9 defines the Physical Requirements, including enclosure and connector requirements.

Clause 10 defines the Electrical Requirements, including line driver and receiver requirements and terminator requirements.

Clause 11 defines the Logical and Command Requirements, including the requirements that apply to the various SCSI data phases.

Clause 12 defines the Target Device Requirements.

Clause 13 defines the Initiator Device Requirements.

Clause 14 defines the Requirements that apply to specific Device Types.

Clause 15 defines the effect on existing standards.

X3 Technical Report for Information Technology -

Profile for SCSI Components Used in High Availability Environments

1. Scope

This technical report defines various methods for the use of parallel SCSI components in systems where high system availability is obtained by the use of multiple hosts and multiple devices connected by one or more parallel SCSI interconnects.

2. References

X3's technical reports are not Standards and thus do not contain binding requirements. Therefore they do not contain normative references. Any statements herein that appear to contain requirements, including instances of sentences with the verb "shall", are advisory in nature. Not following these advisory statements will likely result in implementations that do not accomplish the stated goals of this technical report.

Further details about the use of keywords are contained in clause 3.3.

3. Definitions, Abbreviations, Keywords, and Conventions

3.1 Definitions

3.1.1 High Availability

A feature, which may be incorporated into a computer system, which provides the ability to access user data and the ability to process that data in the event of a failure of a single component of the system. When used as an adjective to modify the description of an object, this term means that the object has the characteristics required to provide such data access and processing availability. For example, a high availability computer system is a computer system with a high ability to access user data and process that data in the event of a failure. A high availability device is a device that is suitable for use in such a system.

3.1.1 Computer System

A collection of hardware and software components that together form a system that performs work for a user by way of executing a program of instructions that describes the work.

3.1.2 SCSI System

A computer system that contains at least one host, at least one appliance, and at least one SCSI domain arranged so that the host and appliance are within at least one domain in common.

3.1.2 Small Computer System Interface

[see if there is a suitable official definition] A method of connecting parts of a computer system. When used as an adjective to modify the description of an object, this term means that the object uses SCSI in the context of its host-appliance connection. For example, a SCSI controller must use SCSI as its host-appliance connection but may use ATA to connect to its storage units.

3.1.3 Host

The part of a SCSI system that executes user programs. A host issues I/O requests to one or more appliances in order to store and retrieve data. A host may be connected to an appliance by one or more SCSI busses. A host usually contains an application client, and usually performs the initiator role on a SCSI bus.

3.1.4 Multi-host System

A SCSI system that contains more than one host.

3.1.5 Appliance

The part of a SCSI system that services I/O requests. An appliance may be connected to one or more hosts by one or more SCSI busses. An appliance usually contains an application server, and usually performs the target role on a SCSI bus.

3.1.6 Host-Appliance Connection

A connection between a host and an appliance in a SCSI system. The host-appliance connection is used to transfer data between the host and the appliance. There may be one or more host-appliance connections between any given host and any given appliance in a system.

3.1.7 Device

A host or appliance that is connected to a service delivery subsystem and supports an SCSI application protocol.

3.1.8 Component

Hardware (e.g. connector or terminator) or software (e.g. operating system device driver program) in a computer system. A SCSI component is one that meets the requirements of the SCSI standard and is connected to or is part of the SCSI host-appliance connection.

3.1.9 Controller

An appliance that contains one or more storage units. A controller may provide the high availability feature for the appliance part of the SCSI system.

3.1.10 Storage Unit

A part of a computer system used to store data. A storage unit may be contained within an appliance and thus may not itself connect to a SCSI bus.

3.1.11 Target (SAM)

An SCSI device which receives SCSI commands and directs such commands to one or more logical units for execution.

3.1.11 Initiator (SAM)

An SCSI device containing application clients which originate device service and task management requests to be processed by a target SCSI device.

3.1.11 Failure

The condition in which a device or component is unable to perform its normal function.

3.1.12 Normal Operation

The mode of operation of a SCSI system in which no failures are present and the SCSI system provides the ability to access user data and the ability to process that data.

3.1.13 Failover Operation

The mode of operation of a high availability SCSI system in which one or more failures are present but the SCSI system continues to provide the ability to access user data and the ability to process that data. The performance of the system may be reduced in this mode of operation.

3.1.14 Failover

The process of recovering from the failure of a device or component by transferring its workload to one or more other devices or components.

3.1.15 Host Failover

An event that occurs in the case of the failure of one of a set of redundant hosts. The failure of the host causes its workload to be transferred to one or more of the other hosts.

3.1.16 Appliance Failover

An event that occurs in the case of the failure of one of a set of redundant appliances. The failure of the appliance causes its workload to be transferred to one or more of the other appliances.

3.1.17 Controller Failover

An event that occurs in the case of the failure of one of a set of redundant storage subsystem controllers. The failure of the controller causes its I/O load to be transferred to one or more of the other controllers.

3.1.18 Standby Mode

An operation mode of a device or component in which, during normal operation, the device or component is available for use but does not participate in supporting the workload of the system. During normal operation the device or component does not contribute to system performance. During failover operation the device or component is used to compensate for the failure or failures.

3.1.19 Active Standby Mode

An operation mode of a device or component in which, during normal operation, the device or component is in use and is used to support the workload of the system. During normal operation the device or component contributes to system performance. During failover operation the device or component is used to compensate for the failure or failures.

3.1.20 Device Plugging

The addition or removal of a device from a storage subsystem. In some cases this may be done with the system active, while in other cases the system must be idle or powered off before the device can be plugged.

3.1.21 Y Cable

A SCSI cable that provides a stub connection in mid-cable.

3.1.22 Console

The mechanism used by human operators to communicate with a system component for control purposes including startup, configuration, and other operations that require low-level access to the component.

3.1.23 SCSI Address Space

The set of SCSI IDs defined by the data signal lines in a given parallel SCSI implementation. For example, a narrow SCSI bus has eight data signal lines and can therefore address eight device IDs, and the SCSI address space is exactly those eight device IDs.

3.1.24 Logical SCSI Bus

One or more physical SCSI busses connected so that as a set they allow all the devices in a SCSI address space, and only those devices, to communicate.

3.1.25 Bus Segment (EPI)

A set of conductors and connectors that attain signal line continuity between a set of one or more drivers, one or more receivers, and exactly two terminators for each signal. The terminators are at the extreme ends of the bus segment and define its extent.

3.1.26 Terminator (EPI)

Interconnect components that form the ends of the transmission lines in bus segments.

3.1.27 Bus Path (EPI)

The electrical connection directly between the two terminators in a bus segment.

3.1.28 Stub (EPI)

Any electrical path in a bus segment that is not part of the bus path.

3.1.29 Stub Connection (EPI)

The point where a stub meets the bus path.

3.1.30 Bus Segment Connector (EPI)

Any connector used to create a bus segment. Bus connectors are defined by both their function and their physical placement.

3.1.31 Bus Path Connector (EPI)

Any connector used to provide part of the bus path. A functional description.

3.1.32 Stub Connector (EPI)

Any connector used to provide part of a stub. A functional description.

3.1.33 Device Connector (EPI)

Any connector physically part of a device. A physical placement description.

3.1.34 Cable Connector (EPI)

Any connector that is physically part of a cable assembly, attached to backplanes, or other non-device connectors. A physical placement description.

3.1.35 Terminator Connector (EPI)

Any connector physically part of a terminator. A physical placement description.

3.1.36 Enclosure Connector (EPI)

Any connector that is physically part of an enclosure. A physical placement description.

3.1.37 Bus Expander (EPI)

A component that couples bus segments together within a domain.

3.1.38 Service Delivery Subsystem (SAM)

Subsystem through which clients and servers communicate.

3.1.39 Service Delivery Port (SAM)

Device-resident component of the service delivery subsystem. This object may contain hardware and software that implements the protocols and interface to the interconnect subsystem.

3.1.40 Interconnect Subsystem (SAM)

A set of one or more physical interconnects that appear to a client or server as a single path for the transfer of data between SCSI devices.

3.1.41 Software

A program of instructions that defines the implementation of a protocol. The term includes firmware permanently contained within the wiring of a component, microcode contained in devices, and driver software contained in hosts.

3.1.42 Boot Software

Software used during the initial power-on period of the operation of a system. Boot software includes programs commonly termed console code, host adapter microcode, device diagnostics, power-on self tests, boot sequences, etc.

3.1.43 Run-time Software

Software used during the normal operation of a system.

3.2 Abbreviations

GByte	gigaByte
HA	high availability
MByte	megaByte
SCSI	Small Computer System Interface

3.3 Keywords

Several keywords are used to distinguish between different levels of requirements, as follows.

- "may" means "is allowed to",
- "should" means "is recommended to", and
- "shall" means "is recommended to in order to accomplish the goals of this technical report".

3.3 Editorial Conventions

The Small Computer System Interface is described in standards including several versions and a number of individual documents. The original Small Computer System Interface Standard, X3.131-1986, is referred to herein as SCSI-1. SCSI-1was revised resulting in the Small Computer System Interface -2 (X3.131-1994), referred to herein as SCSI-2. The set of SCSI-3 standards are collectively referred to as SCSI-3. The term SCSI is used wherever it is not necessary to distinguish between the versions of SCSI.

This technical report is written with the understanding that it is for use with SCSI components as described in the set of SCSI Standards.The SCSI architecture, as well as references to all the relevant standards, is described in the SCSI Architecture Model (SAM) (X3T10-994D). SAM defines the functional partitions and specifies a model for SCSI I/O system and device behavior which applies to all SCSI interconnects, protocols, acces methods and devices. The parallel SCSI bus is described in a pair of standards. These are the SCSI Parallel Interface (SPI) (X3T10-885D) and the SCSI Interlocked Protocol (SIP) (X3T10-856D). Other features of SCSI are described in other related standards.

Certain words and terms used in this technical report have a specific meaning beyond the normal English meaning. These words and terms are defined in the glossary. Lower case is used for words having the normal English meaning.

In the figures, lower case letters are used to identify component and given objects. Upper case letters are used to identify compound objects.

3.4 Numeric Conventions

Digits 0-9 in the text of this technical report are decimal values.

Large numbers are not separated by commas or spaces (e.g., 12345; not 12,345 or 12 345).

4. Objects and Object Notation

The SCSI Architecture Model standard defines the SCSI architecture in terms of objects. The object s are abstractions encapsulating a set of related functions, data types and other objects. Certain objects, such as an interconnect, may correspond to a physical entity while others, such as a task, may only exist conceptually. That is, although such objects exhibit a well-defined, observable set of behaviors, they do not exist as separate physical elements.

An object is a container that may enclose single entities and other objects. For example, an SCSI device may contain logical units. A logical unit may have tasks, a task set and so forth. An object that includes other objects is called a compound object.

The object notation is used to define a grammar that allows a given configuration to be parsed. Parsing a configuration allows a verification of whether the system meets the configuration rules defined by the grammar. A given configuration may be described by a tree structure consisting of a root and one or more leaves. The leaves are either components, such as "terminator", or givens, such as "user data", while the root is the basic descriptor of the system, such as "SCSI high availability computer system".

4.1 Notation for Objects

The notational and graphical conventions for specifying objects are described in SAM. This notation is used to define a grammar that describes the relationship between objects defined in SAM, objects defined in EPI, and objects defined in this technical report.

4.2 Object Definitions from SAM

The following objects and their relationships are defined in SAM. This technical report does not make use of all of the enclosed objects, and the unused objects and relationships are not included here.

SCSI Domain = 2{SCSI Device} + Service Delivery Subsystem

Service Delivery Subsystem = 2{Service Delivery Port} + Interconnect Subsystem

SCSI Device = [Initiator | Target | Target + Initiator] + 1{Service Delivery Port}

Service Delivery Port = Implementation-specific hardware and software

Initiator = 0{Application Client}

Target = 0{Logical Unit} + Logical Unit 0 + 1{Target Identifier} + Task Manager

4.3 Object Definitions from EPI

The following objects and their relationships are defined in EPI. This technical report does not make use of all of the enclosed objects, and the unused objects and relationships are not included here.

[stuff here from EPI]

4.4 Object Definitions for HAP

The following objects and their relationships are defined for use in this technical report.

4.4.1

Several new concepts are introduced.

1. host
2. appliance
3. storage unit
4. host-appliance connection

5.

A term is needed to differentiate between a device that primarily acts as an initiator and a device that primarily acts as a target. These are termed hosts and appliances, respectively. A host is typically a computer. An appliance is typically a disk drive or a tape drive. Both are SCSI devices, and both may take on either initiator or target roles. The host-appliance connection is the communication path between the two. A storage unit is a part of the SCSI system used to store data, and may or may not be connected directly to the host-appliance connection.

4.4.2 Generic Computer System

In general, a computer system provides one or more units of user capability, where a user capability is the ability of a user workload that generates a demand for I/O to be satisfied by a supply of I/O data and where the connection between the demand for I/O and the supply of I/O data is by a user I/O path. The user workload is in the form of a program that is executed by a host, the user data is stored on an appliance, and the user I/O path consists of a host-appliance connection. This configuration is described by the following grammar. The grammar defines a tree structure with the computer system at the root and two types of leaves. A component is a physical [thing] that may fail. A given is a conceptual [thing] that is provided by the user. A component may suffer a physical failure, while a given is assumed to be perfect. For example, a computer system is not able to prevent a user from providing incorrect data, an erroneous program, or an incorrect device name.

Grammar for Generic Computer System:

generic computer system (root) = 1{generic user capability}

generic user capability = user capability + subsystem

user capability = computer system + user mapping

computer system = user workload + user data + user I/O path

user mapping (given) = mapping between user requests and the functionality of the computer system

subsystem = host + appliance + host-appliance connection

user workload (given) = demand for I/O

user data (given) = supply of I/O data to satisfy demand for I/O

user I/O path (given) = logical connection between a specific user workload and the user data required to service that workload

host (component) = executes user workload + creates demand for I/O over host-appliance connection

appliance (component) = services I/O requests + supplies I/O data over host-appliance connection

host-appliance connection (component) = physical connection between host and appliance

A configuration that may be parsed to show that it is a valid computer system is shown in Figure 1. An invalid configuration is shown in Figure 2.

[figures to show parsing of a physical configurations]

Figure 1.

Figure 2.

4.4.3 SCSI High Availability Computer System

This technical report describes a specialized type of computer system. In addition to the basic features of a generic computer system, a SCSI high availability computer system is a computer system that uses one or more SCSI interconnects to provide user capability, and additionally uses the SCSI interconnect(s) to provide the user with the assurance that the failure of a single component in the computer system will not cause the loss of the user capability or capabilities.

The grammar below allows a test to be made as to whether a given system meets the SCSI high availability requirement. The requirement is that the failure of any single component must not cause the failure of the system. [There are some unresolved difficulties in defining the geometry of a complete system below.]

Grammar for SCSI high availability computer system:

SCSI high availability computer system (root) = 1{SCSI high availability user capability}

SCSI high availability user capability = redundancy function(SCSI user capability)

SCSI user capability = user capability + SCSI subsystem

user capability = computer system + user mapping

user mapping (given) = mapping between user requests and the functionality of the computer system

computer system = user workload + user data + user I/O path

SCSI subsystem = SCSI host + SCSI appliance + SCSI host-appliance connection

redundancy function = a function that tests for redundancy of its arguments. Refer to Clause 8.

user workload (given) = demand for I/O

user data (given) = supply of I/O data to satisfy demand for I/O

user I/O path (given) = logical connection between a specific user workload and the user data required to service that workload

SCSI host = SCSI device that executes user workload + creates demand for I/O over SCSI host-appliance connection

SCSI appliance = [SCSI disk | SCSI tape | SCSI controller] + provides user data and services I/O demand over SCSI host-appliance connection. See clause 4.4.3.

SCSI host-appliance connection = connection path between host and appliance within a given SCSI domain

SCSI device (component) = [initiator | target | target + initiator] + 1{service delivery port}. For the purposes of this technical report a SCSI device is considered a single component with no internal structure. Refer to Clauses 12, 13, 14.

SCSI domain = 2{SCSI device} + service delivery subsystem

service delivery subsystem = 2{service delivery port} + interconnect subsystem. Refer to clause 9.

service delivery port = Implementation-specific hardware and software. For the purposes of this technical report a service delivery port is considered an integral part of a SCSI device.

interconnect subsystem = 1{bus segment} + 0{bus expander}

bus segment = 2{terminator}2 + 1{bus path}1 + 0{stub}

bus path = 1{conductors} + 0{bus path connector}

conductor (component) = wire or other electrical circuit that connects exactly two connectors

stub = various stub geometries. For the purposes of this grammar the presence or absence of one or more stubs on a bus segment is not a factor. Stub issues are discussed in clause 10.

bus path connector (component) = connector + provides part of the bus path

bus expander (component) = ...

terminator (component) = ...

connector (component) = ...

[difficulty of differentiation between functional descriptions and physical descriptions starts here]

??? = ... + 1{device connector}

??? = ... + 1{cable connector}

??? = ... + 1{terminator connector}

??? = ... + 1{enclosure connector}

device connector = connector + ...

cable connector = connector + ...

terminator connector = connector + ...

enclosure connector = connector + ...

enclosure = ???

A configuration that can be parsed to show that it is a valid SCSI high availability system is shown in Figure 3. An invalid configuration is shown in Figure 4.

Figure 3.

[figure 3 to show parsing of a physical configuration]

Figure 4.

[figure 4 to show parsing of a physical configuration]

4.4.4 Controllers

The internal structure of an appliance may be important depending on whether it is a traditional disk or tape (or other) device or is a controller attached to one or more storage units. While a SCSI disk or SCSI tape is a component, it does have internal structure. This structure is exposed here only to assist in understanding the concept of controller.

SCSI disk (component) = SCSI disk model + 1{storage unit}

SCSI tape (component) = SCSI tape model + 1{storage unit}

storage unit (component) = a [thing] that is used to store data

controller = [SCSI disk model | SCSI tape model | SCSI controller model] + 1{internal subsystem}

internal subsystem = [SCSI high availability computer system | undefined subsystem]

undefined subsystem = 1{storage unit} + optional behavior

optional behavior (given) = any computer model other than the SCSI high availability computer system model

A configuration that can be parsed to show that it is a valid controller is shown in 5.

Figure 5.

[picture to show parsing of a physical configuration of a controller]

5. High Availability Systems

This technical report describes the requirements for building high availability SCSI systems based on parallel SCSI bus hardware. The purpose of such a system is to maximize the availability of user data in the presence of various failures. These requirements are addressed from the electrical, SCSI device, and software perspectives.

The basic feature of a high availability system is that it continues to operate after the failure of any single component in the system. In general this is accomplished by the use of redundancy of system components and data. In most cases a high availability system has two or more appliances, each with a copy of the users' data stored on non-volatile media, has two or more hosts any of which can execute user application programs., and has two or more connections between the hosts and the appliances. This is not meant to imply that data mirroring is the only way to implement a high availability system--RAID storage or other approaches may be used--but conceptually there is a redundant copy of everything in the system.

Depending on the sophistication of such a system the redundant components may be used in various modes.

- If the system supports the standby mode of operation, the redundant components are only used in the case of a failure. When a failure occurs the appropriate redundant components are used to compensate for the failure. In this mode there is cost associated with the components that are idle during normal operation.
- If the systems supports the active standby mode of operation, the redundant components are used during normal operation. When a failure occurs, the system continues in operation but suffers degraded performance because the remaining components must absorb a larger workload. In this mode there is a cost associated with the loss of performance after a failure.

An additional feature of high availability systems is that they exhibit good design practice in all aspects. This includes the use of components that meet high quality and performance standards, software that performs well under a wide range of system workloads, and hardware that meets certain requirements related to ease of service and maintenance.

The reliability or quality of the user-supplied programs and data are not a factor in this model, and are regarded as given. The goal is to provide a system with the capability of providing high availability that the user may take advantage of.

6. Multi-Host SCSI Systems

There are many approaches to the construction of high availability SCSI systems. This technical report describes systems that use parallel SCSI, but the generic requirements stated herein may also be applied to other interfaces. Among systems using parallel SCSI there are many levels of availability.

6.1 Levels of Availability

6.1.1 Device Level Availability

- System is proof against failure of any single SCSI device component
- SCSI devices may be hot plugged

6.1.2 Component Level Availability

- System is proof against failure of any single component
- Components may be hot plugged

[

One may construct a dual host single disk system with a single SCSI bus and claim that it offers high availability. This claim would be valid in an environment where the reliability of busses and disks is enough to deliver the required level of availability to the user.

At the other extreme one may construct a system with full redundancy and fast automatic failover of every component. This approach would be capable of delivering the highest level of availability to the user.

]

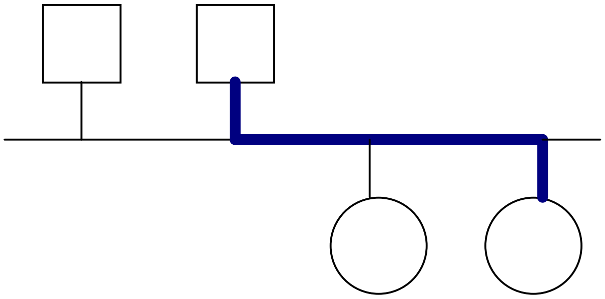
6.2 Single Bus Systems

The most restricted case of a SCSI high availability computer system is if the host-appliance connection is in a domain constructed from only a single bus segment [or certain configurations using bus expanders?]. In these cases a failure of the bus segment causes the domain to fail, which causes the host-appliance connection to fail, which causes the system to fail.

While it is debatable whether such a restricted configuration should qualify as a high availability system, in many practical situations it may be acceptable, particularly if the SCSI bus is considered very reliable. Clearly in such a system every possible effort should be made to ensure that connectors remain securely attached, that cables be routed so as to not be tripped over, etc. Note that if active SCSI bus terminators are used as required by EPI, or if active circuits are used to extend the length of the SCSI bus, these active devices must be accounted for in the high availability strategy.

A configuration of this type is shown in Figure 6.

Figure 6.



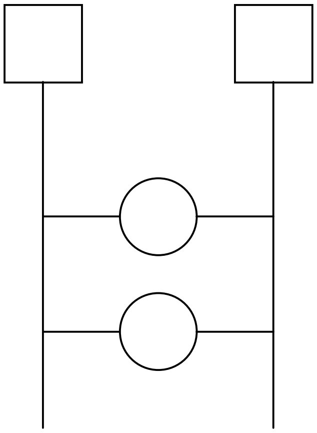
6.3 Multiple Bus Systems

Another case is if the host-appliance connection has a second redundant copy. This can be handled by supplying redundant physical SCSI busses, in which case the system may continue operation using the redundant bus.

From the practical viewpoint, consider that many SCSI devices do not support multiple bus connections.

A configuration of this type is shown in Figure 7.

Figure 7.



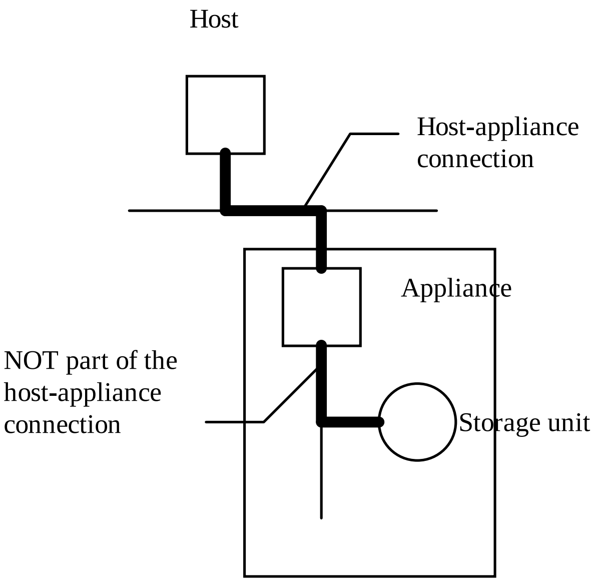
6.4 Controllers

SCSI high availability computer systems may have more than one SCSI bus. This may occur if the appliance is a storage subsystem containing a controller and one or more storage units. A controller connects to a host-appliance connection in the same way as a SCSI disk or SCSI tape but has internal structure that may include more than one disk or tape storage unit.

Depending on the design of the controller, one SCSI bus may be used for the host-appliance connection and another for the connection of the controller to the storage units. The requirement for the busses may be different. For example, the host-appliance connection is required to support hot plugging, but the controller may be constructed so that hot plugging support is not needed on the storage unit bus. For this reason the distinction between the SCSI bus used for the host-appliance connection must be carefully distinguished from other SCSI busses in the system.

Figure 8 shows a typical controller and its internal structure.

Figure 8.


Host
Host-appliance connection
Appliance
NOT part of the host-appliance connection
Storage unit

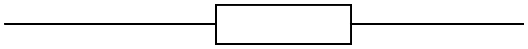
6.5 Domains

The host-appliance connection exists within a SCSI domain. The implementation details of the domain are not specified. This allows the host-appliance connection to be implemented using SCSI bus technology that allows the high availability system to continue operation in the event of a failure of one of the parts used to implement the host-appliance connection. For example, bus expanders may be used to connect multiple separate bus segments in a configuration that implements a single domain containing a host-appliance connection. Certain bus segments may fail without causing the system as a whole to fail.

[Something about power supplies...]

A SCSI domain consisting of two SCSI bus segments and a bus expander is shown in Figure 9.

Figure 9.



6.6 When a Component Fails

By using more than one host in one or more domains it becomes possible to move I/O traffic from one host to another. By using more than one appliance on the host-appliance connection bus it becomes possible to maintain multiple copies of data. These two features allow host failover to occur in case a host fails, and appliance failover to occur in case an appliance fails. To allow these failover operations to occur, the requirements described in this technical report must be implemented in the devices, the software, and the system as a whole.

Note that host failover is different from controller failover, which describes the way that a controller (usually a RAID controller) failure is handled by a system. Controller failover is usually implemented inside a controller cabinet because to obtain high failover performance it is usually necessary that there be a direct high-speed connection between the controllers.

A requirement is that a high availability system must support the removal and insertion of devices from the host-appliance connection SCSI bus while the system is in operation. This may be needed for normal service or in the case of a failure. In either case the system must continue to operate, even if only at a degraded level of performance.

If the high availability system supports an active standby mode of operation, the multiple hosts and multiple appliances may be actively shared. For example, two or more hosts both may be performing I/O operations to a single disk. Coordination between the hosts is needed to prevent data corruption when using this mode of operation. This coordination may be done using communication over the SCSI bus or by an alternate path such as an ethernet connection between the hosts.

6.7 Other Considerations

There are a number of system issues that must be considered to achieve a high availability system, including the failover of network traffic, shared access control, and management of the many issues related to system security, device naming, and shared device access control. These issues are not directly related to the use of the SCSI bus and are beyond the scope of this technical report.

This technical report recognizes that the requirements for a device operating as a host are different from those of a device operating as an appliance, and that diagnostic software has a slightly different set of requirements from boot software or run-time software. Separate clauses in the text are used to describe these and other special cases.

Because the standards that describe SCSI are updated on independent schedules it may be difficult for a device to identify with full detail the exact version number for every standard applicable in a given situation. It is recommended that device vendors identify their product as conforming to the versions of the applicable SCSI standards that were current as of the date when the design was finalized. By using this approach a customer may identify the state of the SCSI standard as it applies to the device in question.

In particular, the SCSI system shall conform with all requirements of the latest revision of the SCSI Parallel Interface standard (SPI). Revision 15a is current as of the date of this technical report.

Specific additional exclusions and expansions to the SCSI standard are described in this technical report.

7. Fundamental Requirements

The basic requirement for SCSI high availability computer systems is that the SCSI devices must be capable of coexisting in the SCSI domain without interference.

This requirement is met by ensuring that at the highest level the system provides

- support for multi-host operation,
- support for hot plugging of SCSI device components or all components,
- high reliability at the component hardware level as a result of good design, and
- good system-wide responsiveness to failure situations.

The SCSI system, including all components, shall conform with all mandatory requirements of the latest revision of the SCSI standard extant at the time of manufacture.

8. SCSI High Availability User Capability Requirements

SCSI high availability user capability = redundancy function (SCSI user capability)

This clause describes requirements for SCSI high availability user capability. These are based on the need for redundancy of SCSI device components or all components.

8.1 Generic High Availability User Capability Requirements

The basic high availability user capability requirement is to support hot plugging of SCSI device components or all components, depending on the desired level of availability.

The requirement for hot plugging implies certain related enclosure considerations.

Mode page setting, logical unit reservations, ID assignments, and the use of bus resets must all be coordinated between hosts

8.2 Specific High Availability User Capability Requirements

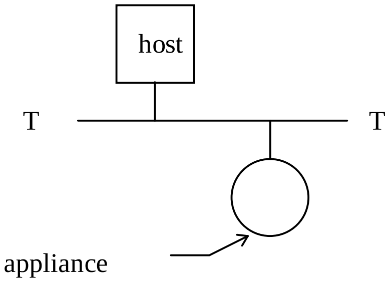
8.2.1 Hot plugging of SCSI Device Components

This requirement may be met by any method that allows a SCSI device to be removed from or added to a system configuration while the system is in active use.

8.2.1.1 Bus Configurations

One method of meeting the generic requirement is to use a simple configuration as is shown in Figure 10, to use devices that meet the Case 4 hot plugging requirements, and to use software that supports Case 4 hot plugging without causing any interruption to normal I/O traffic on the SCSI bus. This method is mainly suitable for backplane situations and may be too restrictive for larger environments.

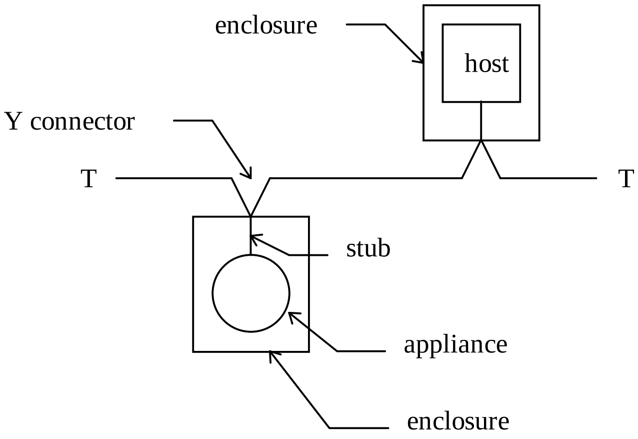
Figure 10.



8.2.1.2 Enclosure with Single External Connector and External Y Cable

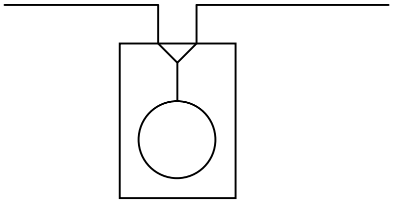
A similar configuration may be constructed for a cabled environment using Y cables. Each device is in a small enclosure with a single connector and a short stub length. This is shown in Figure 11.

Figure 11.



8.2.1.3 Enclosure with Two External Connectors and Internal Y Cable

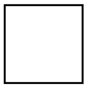
Figure 12.



8.2.1.4 Controller Configurations

Another method of meeting the generic requirement that may be suitable for larger environments is to use "Y" cables and the single connector option for all devices that need to be hot plugged. For example, it may be suitable in some applications if the storage units are grouped together in two or more enclosures and isolated from the host-appliance connection by a controller. By plugging the controller the storage units are effectively plugged as a group. In that case the enclosure must meet the Case 4 requirements and the rules that apply to "Y" cable configurations. Also, using this approach, a single enclosure may contain two or more separate hot pluggable appliances, each with its own separate connection to the external bus, as is shown in figure 13. It may be difficult to meet the SCSI bus maximum stub length requirements using the "Y" cable approach.

Figure 13.



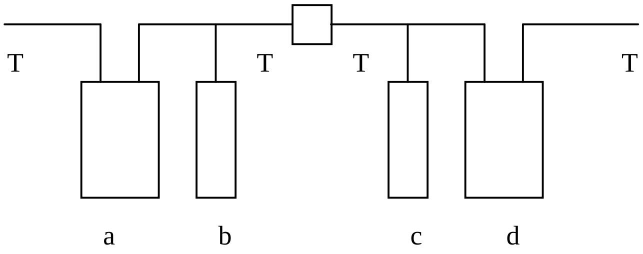
8.2.1.5 Combined Configurations

These methods may be combined on one bus segment. For example, the SCSI hosts may use simple connections to the bus while the storage units are grouped by controller appliances into enclosures.

8.2.1.6 Bus Expander Configuration

Another method of meeting the generic requirement is to use a bus expander that allows the SCSI host-appliance connection (logical concept) to be partitioned into two or more bus segments (physical concepts). For example, in figure 14, the bus expander includes termination on both bus segments. If one bus segment becomes inoperative the other continues to operate. Thus if enclosure "a" is removed from the bus, SCSI device "c" and enclosure "d" are still fully operational.

Figure 14. Use of SCSI Bus Expander



8.2.1.7 Other Configurations

Other methods may also be used to meet the generic requirement.

8.2.2 Hot Plugging Components

Hot plugging of any component in the system.

8.2.3 Enclosure Considerations

It is recommended that the mass storage units in a high availability system be housed in a self-contained enclosure that is separate from the host enclosures. By providing redundant power supplies and cabinet services a storage subsystem can provide SCSI service to more than one host system, with minimal disruption if one host system malfunctions.

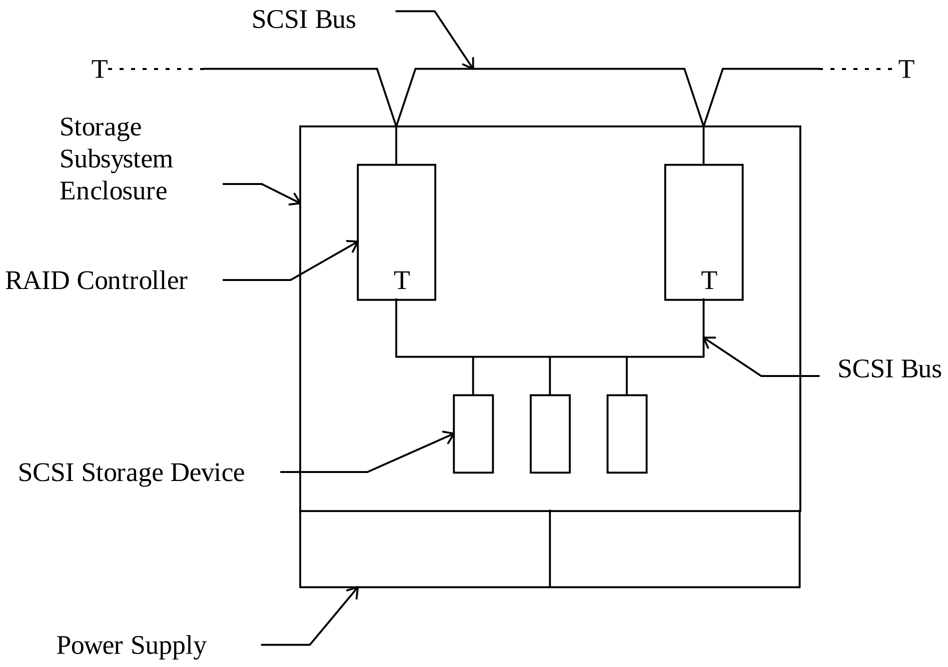
If the redundant storage units are all contained within a single enclosure, it is recommended that dual redundant independent power supplies be provided in the storage enclosure since the maintenance of electrical power to the devices is critical in maintaining high availability. If data redundancy is distributed between more than one storage enclosure then this is not as important.

For example, a high availability storage subsystem enclosure might contain a pair of RAID controller appliances each with a single external SCSI connection, a number of internal SCSI disk drive storage units connected to the controllers, and two power supplies to provide redundancy. The RAID software in combination with the redundant hardware provides data availability. A configuration of this type is shown in Figure 15.

Note that the SCSI bus used for the host-appliance connection is separate from the SCSI bus used inside the subsystem to connect the controller to the storage units. Each of these SCSI busses has its own termination, termination power, and SCSI address space.

In the grammar, power supplies are not considered. [why not?]

Figure 15 Typical Storage Subsystem Enclosure



8.2.4 Coordination of System Resource Sharing Between Hosts and Appliances

SCSI high availability computer systems shall have a mechanism to coordinate the mode page settings on shared devices. The details of mode page coordination are vendor specific. [goal: remove all optional behavior]

Particular examples of mode page values that shall be maintained on a system-wide basis include:

- Default block size
- Read/Write Error recovery page
- Cache control page
- Disconnect/Reconnect page

Targets are not required to maintain mode pages on a per-initiator basis, so all hosts shall be able to operate with the same mode page setup on each appliance. Each appliance may have different mode parameter values, but the values for a given appliance apply across all hosts.

Since the status of a reservation may change upon removal of device power, the hosts shall coordinate the reservations between themselves. The details of reservation coordination are vendor specific. [goal: remove all optional behavior] The persistent reservation option may be used to improve this coordination.

SCSI high availability computer systems shall have a mechanism to coordinate the access to shared data. If this coordination is done using the SCSI bus as a communication method, the RESERVE and RELEASE commands shall be used to protect the shared data.[why not specify use of PERSISTENT RESERVE and release?]

In order to have more than one initiator on the SCSI bus there shall be a cooperative method of handling the SCSI ID assignments on all the devices on the SCSI bus. Either of the following may be used.

- All devices on the bus shall have SCSI IDs assigned in advance by the system administrator and set by switches or jumpers. Each device shall have a unique SCSI bus ID.
- The various levels of SCAM support shall be implemented according to SPI Annex B.

All high availability SCSI devices shall implement at least the "SCAM tolerant" level of SCAM as described in SPI Annex B. Full SCAM support is recommended because it minimizes the chance of device ID conflicts. This is particularly important in environments where hot plugging is used because ID conflicts can cause failures even if the system is functional in all other respects.

The use of BUS DEVICE RESETs shall be coordinated between all the hosts in the system.

Software shall run correctly in all devices on an active SCSI bus segment. In particular, boot software shall not affect active I/O on the bus segment.

9. Service Delivery Subsystem Requirements

service delivery subsystem = 2{service delivery port} + interconnect subsystem

interconnect subsystem = 1{bus segment} + 0{bus expander}

bus segment = 2{terminator}2 + 1{bus path}1 + 0{stub}

This clause describes the requirements placed on the service delivery subsystem. The service delivery subsystem consists of two parts, the service delivery ports and the interconnect subsystems. The service delivery ports are integral to the SCSI devices. The interconnect subsystem is the physical plant required to implement a parallel SCSI interconnect. The interconnect subsystem includes at least one bus segment plus zero or more bus expanders, and a bus segment includes exactly two terminators, exactly one bus path, and zero or more stubs.

9.1 Generic Service Delivery Subsystem Requirements

The basic physical requirement is that the domain should be capable of supporting multiple hosts in a dynamic configuration. The dynamic nature of the system means that it must be possible to make changes to the components that make up the system while the system is operating. This generally means that those changes much be possible without the need to perform objectionable levels of hardware manipulation. For example, it should not be necessary to remove external enclosure cabinet panels in order to obtain the access needed to perform a device hot plug operation. On the other hand, front cabinet doors or covers that are designed to be easily removable are acceptable.

For most installations each bus segment should be expected to be at least several meters in length. This is because the use of multiple hosts and multiple appliances generally implies the use of multiple enclosures, which in turn implies that the interconnecting bus segment must be of some significant length.

The installation should protect against invalid bus segment configurations. It is possible to assemble valid components into domains that don't work. It is desirable that the system have a means of performing an automatic verification of the configuration to avoid this situation. This means that one should not construct a Fast-20 single ended bus segment with a 20 meter length, and that the system should detect it if it happens. Since the SCSI protocol does not support the discovery of invalid configurations, the system should have a method for restricting hosts so that they will not negotiate wide or fast methods of operation (i.e. will always operation in narrow synchronous mode). Control of this behavior should be under the manual control of the system administrator, since automatic methods of backing down to a lower level of performance may mask errors resulting from conditions not related to the configuration in a properly configured system.

The bus segment should support device plugging (the removal or insertion of a device on the SCSI bus) of devices or hosts without disturbing the other devices on the bus.

9.2 Specific Service Delivery Subsystem Requirements

9.2.1 Enclosures

Device enclosures should be designed so that they need not be opened to change the configuration from the single-host configuration to the multi-host configuration. This allows newly delivered enclosures to be used in either environment without the need for a physical configuration process during installation.

In order to allow devices to be removed from the bus segment without interrupting bus activity, the bus path shall consist of a series connection of conductors that provide electrical continuity and bus termination. This shall be maintained before, during, and after a device hot plug operation. To accomplish this requirement, SCSI devices, and enclosures shall conform to one of the two cabling options listed below. Refer to Note 3, SPI (page 8).

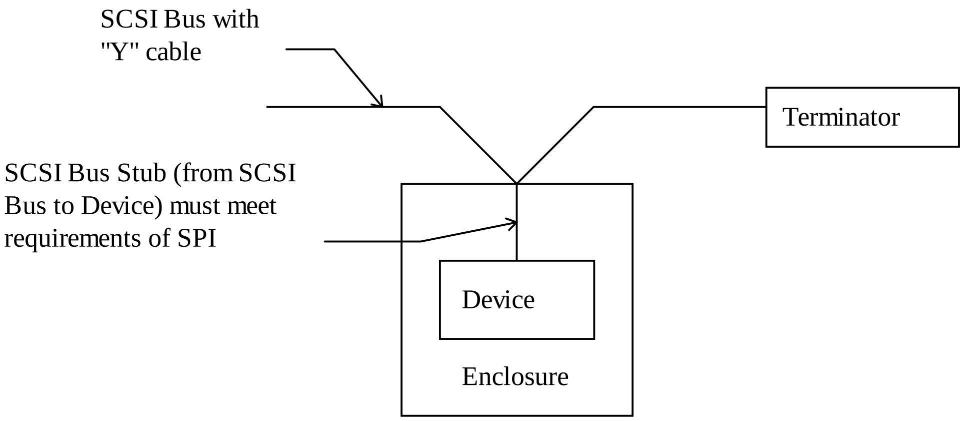
The two options may be mixed on a single SCSI bus as long as the continuity requirement is met.

9.2.2 Connector Options

9.2.2.1 Single Connector Option

If the single connector option is chosen then the enclosure shall be implemented with a single external connector, as shown in Figure 16. In order to remove the enclosure from the bus segment without interrupting bus activity, the bus path shall be equipped with "Y" SCSI cables. Terminators shall not be installed inside the enclosure, but shall be connected directly to the SCSI bus itself, external to all enclosures. [too restrictive] The stub length of the connector and cable inside the enclosure shall meet the requirements of SPI clauses 6.4 and 6.5.

Figure 16.



Because of the SCSI bus stub length restrictions described below, the "Y" cable approach is usually applicable only to those situations where the enclosure contains devices that make separate individual external connections.

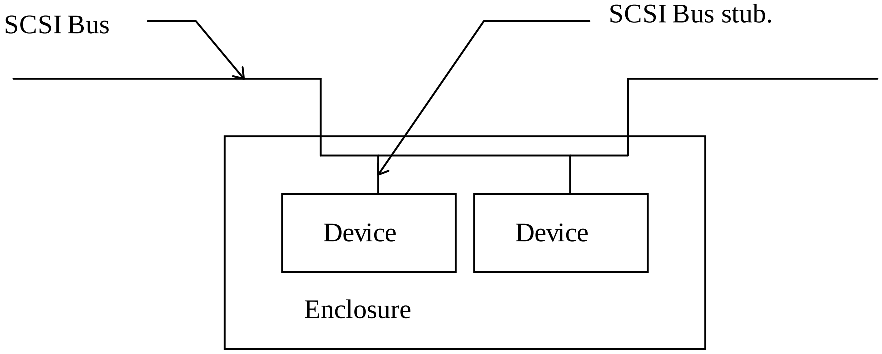
- For example, a storage subsystem with a RAID [add to list of acronyms] controller might have only one external connection to the RAID controller.
- For example, a host computer might only have a single host adapter that makes an external connection.
- For example, a storage subsystem with a pair of RAID controllers might have two external connectors, each intended for use by one controller by itself.
- For example, a storage subsystem consisting of a number of disks on one SCSI bus will probably not use the "Y" cable approach because it is too difficult to meet the stub length requirements.

9.2.2.2 Double Connector Option

If the double connector option is chosen, the enclosure shall be implemented with two external connectors as shown in Figure 17. In this case the bus path enters and exits the enclosure using the two connectors, and the internal wiring is arranged to minimize the stub length caused by the device connection. Such an enclosure shall meet the requirements of SPI clause 5.2.

The use of the two connector option for enclosures allows more internal wiring flexibility. However, they cannot be removed from the domain without disrupting bus activity unless bus expanders are used to isolate bus segments within the domain. Even if bus expanders are not in use, an enclosure suitable for high availability systems may be wired this way if it allows the SCSI devices it contains to be removed without disruption of the bus segment, and if it maintains bus path continuity when a SCSI device is removed from the enclosure.

Figure 17.



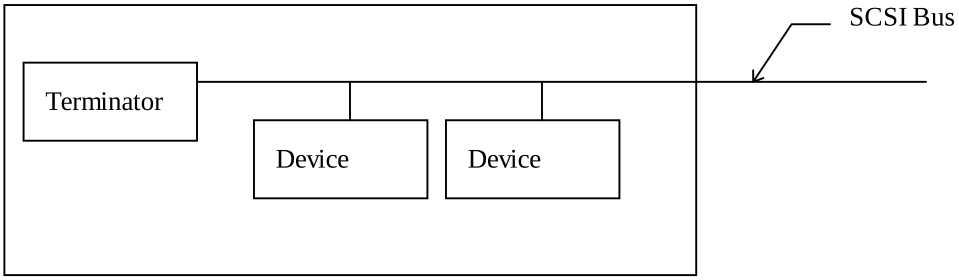
9.2.3 Enclosure with Internal Hosts and Appliances

Many small desktop computer systems are designed using the generic computer system grammar. These computers typically use a combination of internal and external storage. For example, the system disk may be an internal storage unit while the data storage disks and backup devices might be appliances in an external enclosure. In either case the SCSI bus must be terminated properly. If such a system is designed for use in single user environments only, the physical location of the SCSI bus terminators is not important.

It may be desired at some point to integrate a generic computer system such as the above into a SCSI high availability computer system. If the system was designed without regard to the SCSI high availability computer system requirements, then the extensive use of bus expanders may be required in order to provide adequate hot plugging capability. If the system was designed with the SCSI high availability computer system requirements taken into consideration, the integration process is simpler and the use of bus expanders may not be required.

An example of a difficult case is when the system contains an internal terminator on a bus segment that is to be connected to the SCSI high availability computer system bus segment. If it is impossible or difficult to remove or disable this terminator, then the system must be located at the end of the bus segment. If more than two of these systems are to be connected, bus expanders must be used to provide isolation of the bus segments within the domain. This situation is shown in figure 18.

Figure 18.



Other enclosure configurations may also be suitable for use in SCSI high availability computer systems, as long as they meet the above requirements.

9.2.4 Cable Requirements

[Refer to EPI class 1 rules for cable length limits: "derived from field and lab testing"]

The SCSI cable plant shall meet the recommendations of SPI Annexes D and F. Because this technical report is intended to maximize the interoperability of components that are designed for use in high availability systems, and since the SPI Annexes describe a set of requirements that can be expected to support satisfactory operation within certain configuration rules, a specific set of cable length restrictions is useful and necessary.

Since the short cable lengths associated with higher clock speeds may be impractical for high availability systems, it is expected that most such systems will use the differential signalling alternative.

Cable lengths for a complete system shall not exceed the following values.

9.2.4.1 Single Ended Cable Length Requirements

Up to 5 Megatransfers per second:	6 meters
5 to 10 Megatransfers per second:	3 meters
10 to 20 Megatransfers per second:	1.5 meters

9.2.4.2 Differential Cable Length Requirements

All speeds:	25 meters

9.2.4.3 Low Voltage Differential Cable Length Requirements

tbd

9.2.4.4 Backplane Length Requirements

tbd

9.3 Device Plugging

SPI describes four cases in which devices may be removed or inserted from a SCSI bus. For each case the complete applicable requirements are listed. Refer to SPI Annex A. These cases are summarized as follows: [there needs to be a reference to this section in the previous places where plugging is mentioned.]

- Case 1 - Power off during removal or insertion.
- Case 2 - RST signal asserted continuously during removal or insertion.
- Case 3 - Current I/O process not allowed during insertion or removal.
- Case 4 - Current I/O process allowed during insertion or removal.

For high availability systems it is desireable that the operation of removing or inserting a device on the bus cause the smallest possible disruption of bus traffic. Case 1 clearly causes substantial disruption at the system level. Cases 2 and 3 cause less system disruption but because the SCSI bus is stalled for the duration of the removal or insertion operation, they are considered unacceptable for high availability systems.

high availability systems shall support Case 4 device removal and insertion. This is referred to as "hot plugging" in the remainder of this technical report.

[Consider discussion of how to handle "mode change" on plugging of lvd devices; should there be a reset and if so who does it--initiator?]

10. SCSI Electrical Requirements

This clause describes the electrical requirements applicable to all devices connected to the SCSI bus.

10.1 Generic SCSI Electrical Requirements

Since the primary goal of high availability systems is to be able to maintain continuous operation in the event of a component failure, a basic requirement of such systems is that component replacement should be possible while the system is in operation. To support this, the SCSI bus and associated electronics should support hot plugging. Support for hot plugging implies some additional requirements, as follows. Refer to SPI for the specific electrical requirements for supporting hot plugging.

After a component is replaced using a hot plugging operation, power must be re-applied to the component. The component must power-up in a fashion that does not disrupt traffic on the SCSI bus.

Bus termination needs to be external to the enclosure. This allows the enclosure to be removed in the event that it needs to be replaced.

There may be a need for longer pins in some connectors in order to properly sequence the connection and disconnection of the power, ground, and data circuits.

The driver and receiver electronics should be able to withstand the hot plugging operation. The signal lines should be well-behaved during the hot plugging operation. Other devices and in-progress transfers should not be disturbed by the hot plugging operation.

Devices should be tolerant of electrostatic discharge on external connector pins.

10.2 Specific SCSI Electrical Requirements

10.2.1 Termination

All SCSI bus termination devices shall be mounted externally to all enclosures. A device shall not be terminated internally within an enclosure. A device shall not be terminated in such a way that precludes it from occupying any position on the SCSI bus.

Switchable terminators may be used if there is a mechanism for them to be disabled, such as by software or a jumper.

[The following assertion is untrue because a bus extender connects two physical busses into one logical bus. It could say something like: "If a bus extender or converter is located at the end of a bus segment, ...] Since bus extenders and converters (e.g. single-ended to differential) are always located at the end of an electrical bus segment, bus termination shall be provided nearby. This termination shall be external to the extender or converter.

Terminator power shall be supplied as described in SPI clause 7.3, except that "optional internal terminators" shall not be used. Redundant sources of terminator power shall be supplied.

10.2.2 Power Cycling

The driver and receiver electronics should be well-behaved during power cycles. Glitches or other irregular signals shall not be caused on the bus as a result of the application or removal of power to the device during or in preparation for the plugging operation. The bus drivers , receivers, and all other electrical connections to the data, REQ, ACK, and control lines on the device shall maintain the high-impedance state during power-up cycles until the drivers are enabled and during power-down cycles after the drivers have been disabled. Refer to SPI clauses 7.1.2 and 7.2.2.

After power is initially applied to a device after it is plugged, the unit attention flag shall be set for each initiator on each valid logical unit.

10.2.3 Hot Plugging

In order to ensure glitch-free insertion and removal of devices onto to, or off of the SCSI bus, SCSI devices shall conform to SPI paragraph A.4, "Current I/O Process Allowed During Insertion or Removal".

The system software shall prevent bus activity to the device that is to be plugged (the device becomes inactive on the SCSI bus), and the system hardware shall guarantee that the device power and ground connections are made before the SCSI signal lines, in conformance with the requirements of SPI paragraph A.4.

11. SCSI Logical and Command Requirements

This clause describes how SCSI messages and commands are used in a high availability environment.

11.1 Generic SCSI Logical and Command Requirements

The SCSI bus reset mechanism is not well suited for use in a multi-host environment because it is extremely disruptive to in-progress I/O operations. In a multi-host environment targets must maintain some data structures on a per-initiator basis, and must perform message phase actions so as to cooperate with other devices on the bus.

11.2 Specific SCSI Logical and Command Requirements

11.2.1 Device Identification

The device ID mechanism used by the bus should be fully supported by the host's operating system. If the host software requires fixed bus IDs, then the devices on the bus should implement fixed bus IDs. If the host software supports dynamic device addressing, the devices on the bus should implement dynamic addressing.

In order to be able to uniquely identify a device regardless of where it is in a configuration, or after swapping, or in the case of multiple access paths, high availability SCSI devices shall support the vital product data unit serial number page.

Every device shall have a unique identification string, consisting of the vendor name and model name from the standard Inquiry data, concatenated with the device's serial number. If the device implements SCAM, the SCAM identifier string will serve this purpose.

11.2.2 Resets

Since general bus resets can be extremely costly in terms of performance across the entire system, they should be issued only as a last resort. SCSI Targets shall not assert the SCSI RST signal.

Since specific device resets (BUS DEVICE RESET) may interfere with device activity that was started by another host, device resets shall be sent only by hosts. Hosts that issue BUS DEVICE RESETs shall coordinate their use between themselves.

11.2.3 Renegotiation of Bus Options

All initiators should renegotiate any bus options (e.g. wide SCSI) with any device that may have been replaced or power cycled since it was last used. This should not be done on every command, but after a host determines that a bus event has occurred.

Hot plugging operations shall be done in such a way that any required renegotiations are performed.

All initiators should renegotiate any bus options such as wide or synchronous before issuing any command resulting in data transfer, including the INQUIRY or REQUEST SENSE commands. Refer to SIP clauses 8.2.12 and 8.2.15 for additional rules.

Host adapter devices shall offer a mode of operation in which the adapter does not negotiate for anything higher than narrow synchronous operation. This maximizes the chance that the system will work even in the case of invalid cable configurations such as mixed wide and narrow cables and all wide devices. (Note that this does not help in configurations where the maximum bus length is exceeded.)

Upon host bootup, the console software of high availability SCSI initiators shall negotiate data transfer width (using the WDTR message) and synchronous data transfer speed and offset (using the SDTR message) before attempting to execute any SCSI command that requires a data-in or data-out phase (including the INQUIRY command) to any target. Initiators in bootup mode shall assume that any device on the bus may have been powered on, reset, or have changed to a fast or wide mode since the last time it was used by the initiator.

This DOES NOT mean that these negotiations should be done before each command, rather the console (NOT runtime drivers) should do this before the first I/O during the boot or shutdown sequence.

11.2.4 Message and Command Interpreter Validity

All devices shall handle all possible incoming messages at all times. In particular, console software used during the host initialization process shall implement the complete message protocol because other traffic may be active on the bus during host initialization.

Devices shall implement all the mandatory SCSI commands for the device type they report. Optional commands that are not implemented should be handled properly according to the SCSI standard.

Devices shall power up and complete their self tests properly regardless of the state of the SCSI bus. This includes cases of no terminator power, no termination, no bus attached, activity on an attached bus, and reset asserted. Devices that do not meet these requirements may experience difficulty if the devices on the bus do not power up simultaneously.

high availability SCSI devices shall return a status of CHECK CONDITION with sense key of ILLEGAL REQUEST for any unsupported command. [difficulty here of what is definition of "device"--does it include initiators?]

high availability SCSI devices shall return a status of CHECK CONDITION with sense key of ABORTED COMMAND/MESSAGE ERROR for any unsupported message. [How many Bytes of an unsupported message should the target swallow before giving up?]

Parity checking shall be enabled. Parity shall be checked on all received data, in any information transfer phase and during the Selection or Reselection phases. Invalid parity shall be handled in accordance with the SCSI standard.

11.2.5 Per-Initiator Data Structures

The SCSI unit attention flag shall be maintained on a per-initiator basis in the device, as required by SAM in paragraph 5.6.5. The unit attention condition shall be generated for each initiator on each valid logical unit whenever the target receives a BUS DEVICE RESET message, a bus reset, or a power cycle. The unit attention condition shall persist on each logical unit for each initiator until that initiator clears the condition on each logical unit.

Targets shall be capable of maintaining separate sense data for each initiator and shall not return BUSY status to any initiator as a result of a pending contingent allegiance condition with any other initiator.

11.2.6 Handling ABORT Message

If an ABORT message is received, the device shall abort the current operation in a manner that does not cause loss or corruption of data. Parity errors shall not be written to the media, and data for which a GOOD status has been returned to the initiator shall be written to the media before processing the ABORT. Targets that require additional time for this buffer flushing operation shall return BUSY status in response to connection attempts. [Question of handling of case where device writes to media before returning status.]

12. SCSI Target Device Requirements

This clause applies to each SCSI device that is operating as a target.

12.1 Generic SCSI Target Device Requirements

[what about linked commands?]

Devices shall properly handle bus resets that may occur at any time. Setup information shall not be carried across a bus reset (except for saved mode parameters as described in SCSI) because a newly added host cannot predict the earlier information.

Devices shall properly support simultaneous hosts. Devices shall be able to accept and process commands from multiple initiators at any bus IDs without hanging the bus, violating the SCSI standard, or crashing or hanging themselves. Sense data should be retained on a per-initiator basis.

Devices shall maintain mode pages on a per-logical unit basis. This is needed because hosts view each logical unit as a separate device.

Devices shall properly handle device reservation (using either RESERVE or PERSISTENT RESERVE) in a multi-host environment. Note that some systems may never issue RESERVE/RELEASE commands. However, since RESERVE/RELEASE provide a convenient mechanism for low level synchronization it is desireable that they be supported by all targets.

Devices shall support tagged commands in a timely manner. Adherence to this requirement has a substantial impact on the overall responsiveness of a multi-host system. Proper behavior is as follows:

1. The device may defer a given command until all the previously queued commands complete, regardless of the time those commands require. (For example, a FORMAT or REWIND command may take considerable time.)
2. The device may reorder commands (within the rules of tagged command queueing) to defer the execution of a given command until a later time.
3. From the time at which a given command could first have been executed, the device shall not defer execution of the command more than one second.

12.2 Specific SCSI Target Device Requirements

12.2.1 Reset Support

high availability SCSI target devices shall not issue SCSI bus resets.

When a target that is holding data in a cache before writing it to non-volatile storage receives a bus reset, it shall write the cache contents to the media before processing the reset.

12.2.2 Initiator Support

Targets in multi-host systems shall be able to accept and process commands from multiple initiators.

Targets shall accept and process commands from initiators located at any bus ID.

Targets shall maintain the following on a per-initiator basis.

- Synchronous negotiated state.
- Width negotiated state.
- Contingent Allegiance state.
- Unit attention flag.

The Unit Attention Condition, as stored in the unit attention flag, shall indicate whether the mode parameters in effect for this initiator have been changed by another initiator, or if the mode parameters in effect for the initiator have been restored from non-volatile memory, or if any of the normal SCSI Unit Attention Condition conditions apply. Refer to SAM clause 5.6.5.

Devices shall either retain sense data on a per-initiator basis and also return the sense data to the correct initiator or shall stop processing during the auto contingent allegiance condition. Refer to SAM clause 5.6.1.

12.2.3 Logical Unit Support

If a target device supports multiple logical units, then mode pages shall be maintained on a per-logical unit basis.

high availability SCSI target devices shall support Tagged Command Queuing. Note that sCSI-2 clause 7.8.2 requires that all comman d received with Simple Queue Tag message prior to a command received with Ordered Queue Tag message, regardless of Inititator, shall be executed before that command with the Ordered Queue Tag Message. This is essential for the correct operation of the queueing algorithm when used in multi-host systems.

high availability SCSI target devices shall support drive-based Bad Block Replacement (BBR) as described in SCSI. This is required in multi-host systems to avoid potential wastage of revectoring resources in the case where two hosts attempt simultaneous revectoring.

high availability SCSI target devices shall implement a reselection retry algorithm that limits the amount of bus time spent attempting to reselect a non-responsive initiator. In order to prevent retries from timing out other devices, high availability SCSI devices shall delay at least 2.4 milliseconds between retry attempts. Targets shall respond to Initiator selection attempts that occur during the 2.4 millisecond delay between retry attempts.

A target having multiple queued commands for an initiator that fails to respond to reselection, including retries, shall abort all commands from that initiator in the queue. The device shall generate a contingent allegiance conditions for the timed-out initiator with a sense key of HARDWARE ERROR and an ASC/ASCQ of SELECT OR RESELECT FAILURE. This allows multi-initiator environments to continue operation with minimal impact. After having aborted all commands for the timed-out initiator, the device shall generate a contingent allegiance condition for the timed-out initiator with a sense key of HARDWARE ERROR (04h) and an ASC/ASCQ value of SELECT OR RESET FAILURE (45/00h).

Target devices that do not support wide SCSI shall respond to initiator attempts to negotiate wide operation by returning the WDTR message in sequence specifying narrow operation rather than sending MESSAGE REJECT. Completing the WDTR sequence rather than rejecting it resets any previously negotiated synchronous data transfer agreement to the default asynchronous mode. This will prevent bus hang conditions due to synchronous/asynchronous mismatch between targets and initiators. Targets shall track the negotiated wide transfer agreements on a per-initiator basis.

In every case where a WDTR message is sent, it should be followed by an SDTR. This method guarantees that the host and device are in agreement with respect to these two modes of operation.

12.2.4 Error Condition Support

Targets shall manage fault conditions by going to the Bus Free state only in those cases where this action is required by SCSI for catastrophic errors. Exception conditions other than those requiring Bus Free per SCSI shall be handled by other means such as going to STATUS phase with a check condition, by sending a message reject, or by retrying the phase as appropriate.

A target device that terminates a WRITE command with a check condition due to parity errors shall not write the associated data to the media. [Question of cache contents.]

12.2.5 RESERVE/RELEASE Support

Targets shall support the RESERVE and RELEASE SCSI commands. These commands allows hosts to allocate devices with exclusive access.

Targets shall support the following mechanisms to clear device reservations:

- RELEASE Command
- BUS DEVICE RESET Message
- SCSI Bus Reset
- Power Down/Remove
- Targets that are reserved by an initiator shall accept and process the following commands received from any initiator. All other commands shall be failed with a SCSI status of RESERVATION CONFLICT.
- INQUIRY
- REQUEST SENSE
- PREVENT ALLOW MEDIA REMOVAL (Prevent bit cleared to 0) (removable devices)
- RELEASE

If the RELEASE command is received from the initiator that has the outstanding reservation, the reservation is cancelled. If the RELEASE command is received from another initiator, the command is failed with a SCSI status of RESERVATION CONFLICT.

13. SCSI Initiator Device Requirements

This clause describes the requirements that apply to SCSI devices that are operating as initiators.

13.1 Generic SCSI Initiator Device Requirements

The high availability configuration rules must be followed.

The SCSI bus is expected to be active at all times, so every software and hardware function must operate correctly even during power cycles and other system-level state changes.

Hosts should minimize the number of bus reset operations that they initiate. This means that a host should attempt to avoid the reset condition during initialization, normal processing, and shutdown. A bus reset should be asserted only when it is determined that no other method of restarting the bus is possible. Prior to resetting the bus the host should coordinate with other hosts on the bus.

Logical units must ensure that adequate command queue space is reserved for cases where multiple initiators wish to communicate at the same time.

13.2 Specific SCSI Initiator Device Requirements

13.2.1 Configuration Rules

It is particularly important that host and host adapter designs comply with the requirements for external bus termination. It is also important to consider the system implications of the choice between the single-connector and dual-connector options.

If these requirements are not met, the SCSI system may be limited to two hosts that cannot be hot-plugged. This is not adequate for a high availability system.

The host shall implement Target Mode operation as a processor device, and shall support all mandatory requirements of SCSI pertaining to the processor device type.

13.2.2 Bus Support At All Times

Because bus activity is expected to continue during the power sequencing, removal, replacement, and reboot procedure on a failed host in a high availability system, there is no distinction between the requirements placed on the console software, the host adapter sooftware, and the normal runtime driver software environment. Every device on a high availability SCSI bus shall meet all the requirements at all times. This is a notable difference from a single-user system where boot-time discrepancies from normal SCSI usage are common.

If a host is halted by an operator command (such as a console halt command) the SCSI bus host adapter shall not stall in a state that prevents the other devices on the SCSI bus from continuing normal operation. In particular, if the Initiator has a connection active, that connection shall be ended either by following the Target's phase and data transfer requests until the next Bus Free condition, or by asserting ATN and sending the ABORT message. If the second alternative is followed, the Initiator must still react to the target until the Bus Free condition.

Even during the period when a system is starting up or shutting down in a multi-host environment, it is still possible for other initiators to select the system as a target. This selection shall be treated as a normal event and handled in such a way that will allow the currently executing boot or shutdown activity to complete without error.

Three optional solutions to this situation may be used: [goal: remove all optional behavior]

- Disable selection as a target in the console or adapter card. This option is most appropriate for adapters that have little intelligence on them.
- Enable selection as a target and support the INQUIRY, TEST UNIT READY and REQUEST SENSE SCSI commands. Use of this option implies that until the host software has completed its boot process, the console microcode shall be able to respond to and process these commands, and shall either completely implement all of the SCSI message protocol or correctly REJECT any unsupported SCSI bus messages received.
- Enable SELECTIONs and return SCSI status of BUSY, and then return the COMMAND COMPLETE message.

13.2.3 Reset Support

The SCSI bus reset signal is extremely disruptive to in-progress bus activity and can be require lengthy recovery activity, particularly in the case of systems with tape drives and highly cached storage subsystems.

high availability SCSI initiators, including system consoles, and adapter microcode, and mainline driver code, shall not issue SCSI bus resets except when the SCSI bus is "hung". The definition of "hung" is system dependent, but the following procedure is recommended.

From the viewpoint of a given initiator the bus may be hung in communication either with that initiator or with another initiator. In either case, the initiator shall use the same procedure to attempt recovery. This approach is taken because otherwise the bus hang recovery algorithm is dependent on the inter-host coordination method.

The initiator that suspects that a target is hung should first issue an INQUIRY command to the target. If the command goes through the required bus phases then the bus itself is assumed not to be hung. If this fails, the initiator should attempt to coax the target to MESSAGE OUT phase by issuing a Clear Queue message, and then send an ABORT message. If that fails then the initiator should attempt to send the BUS DEVICE RESET MESSAGE, if that fails the initiator should communicate with the other cooperating hosts on the bus to determine whether it is ok to issue a bus reset.

It may be better to remove a suspect device from the bus than to attempt on-line recovery. The reset signal shall be used only as a last resort.

A third party reset is a reset that an initiator detects that was generated by another device. The initiator shall be able to recover from a single or repeated third party SCSI bus resets. The initiator should not take longer than 60 seconds to recover from a single SCSI bus reset or from the last of any series of resets.

(Note: The 60 second requirement is intended to define a guideline for the amount of time the SCSI I/O subsystem may take to recover from a bus reset. The actual recovery time for the entire system may be longer than 60 seconds, depending on a number of factors including the number of spindles on the bus, whether failover actions occur and whether or not the file system recovery is fast or slow.)

When a device receives a hard reset, it shall first ensure that all cached data for which good status has been returned is written to non-volatile media prior to processing the reset. If the device is a sequential access device it shall additionally write an EOD to the medium after flushing the cached data, then it shall rewind the media to BOT.

For sequential devices, it is acceptable to return BUSY status while the device is flushing its buffer and rewind operations are complete. The preferred action is to immediately process the command. Accepting the command and disconnecting to wait until the device can process it will result in host timeouts.

If a device receives multiple valid bus resets in succession, it shall process the reset and recover within 250 msec. [need clarification here again re "device" vs "device" as used above in 60 second discussion]

13.2.4 Command Queue Support

Initiators shall not use all the tag queue depth in a device. [does there need to be a standard way to determine the queue tag depth?]

The initiator shall reserve some number of tag queue elements so that other initiators may still send commands (such as INQUIRY) to the device while it is in use by another initiator.

14. SCSI Requirements for Specific Device Types

Certain additional requirements beyond those required in SCSI may be needed in high availability systems.

14.1 Direct Access Device Type Requirements

TBD

14.2 Sequential Access Device Type Requirements

TBD

14.3 Other Device Type Requirements

TBD

15. Effect On Existing Standards

No changes are required in the existing SCSI standard to support multi-host high availability systems. This technical report describes a number of additional restrictions and implementation rules that, when applied in addition to the requirements of the SCSI standard, allow systems to be constructed that have a high degree of tolerance to SCSI component failures.

Future versions of the SCSI standard may be modified to include some of the rules described herein.

Many of the concepts described in this technical report apply to interconnects described in other ANSI standards such as Fibre Channel and SSA. Similar reports applicable to those standards may be appropriate in the future.

16.