[redacted] 79
X3T9.2/92-[redacted]R0
Revision 0

[redacted] proposed
[redacted]

SCSI-3 Architecture Model
(SAM)

April 24, 1992

Secretariat

Computer and Business Equipment Manufacturers Association

Abstract: This document is an working draft of the SCSI-3 Architectural Model. The purpose of the architecture is to define a model for device behavior that is independent of physical interconnect technology.

This is a PROPOSAL [redacted] of X3T9.2, a Task Group of Accredited Standards Committee X3. As such, this is not a completed standard. The contents are actively being modified by the X3T9.2 Task Group. This document is made available for review and comment only.

POINTS OF CONTACT:

X3T9.2 Chair	X3T9.2 Vice-Chair	Technical Editor
John B. Lohmeyer	I. Dal Allan	Charles Monia
NCR Corporation	ENDL	Digital Equipment Corporation
3718 N. Rock Road	14426 Black Walnut Court	334 South Street
Wichita, KS 67226	Saratoga, CA 95070	Shrewsbury, MA 01545
Tel: (316) 636-8703	Tel: (408) 867-6630	Tel: (508) 841-6757
Fax: (316) 636-8889	Fax: (408) 867-2115	Fax: (916) 841-6400
Internet:	Internet:	Internet:
john.lohmeyer@wichitaks.ncr.com	2501752@mcimail.com	monia@starch.enet.dec.com

draft working document SCSI-3 Architecture Model X3T9.2/92-79R0

STATUS OF EDITING:

Rev 0: Draft of glossary and outline

X3T9.2/92-079 R0

1 Forward

With any technical document there may arise questions of interpretation as new products are implemented. The X3 Committee has established procedures to issue technical opinions concerning the standards developed by the X3 organization. These procedures may result in SCSI Technical Information Bulletins being published by X3.

These Bulletins, while reflecting the opinion of the Technical Committee which developed the standard, are intended solely as supplementary information to other users of the standard. This standard, ANS X3.xxx-199x, as approved through the publication and voting procedures of the American National Standards Institute, is not altered by these bulletins. Any subsequent revision to this standard may or may not reflect the contents of these Technical Information Bulletins.

Current X3 practice is to make Technical Information Bulletins available through:

Global Engineering Documents
2805 McGaw
Irvine, CA 92714
(800) 854-7179
(714) 261-1455

2 Scope

The documents under X3T9.2 jurisdiction are:

1. SCSI-3 Block Device Command Set
2. SCSI-3 Stream Device Command Set
3. SCSI-3 Other Device Command Set
4. SCSI-3 Architecture Model
5. SCSI-3 Common Access Method
6. SCSI-3 Interlocked Protocol
7. SCSI-3 Parallel Interface

The original Small Computer System Interface Standard, X3.131-1986, is referred to herein as SCSI-1. SCSI-1 was revised resulting in the Small Computer System Interface - 2 (X3.131-199x), referred to herein as SCSI-2. This standard, the SCSI-3 Interlocked Protocol, and the SCSI-3 Command Set are referred to herein as SCSI-3. The term SCSI is used wherever it is not necessary to distinguish between the versions of SCSI.

3 Referenced Standards and Organizations

4 Glossary and Conventions

4.1 Glossary

This section contains a glossary of special terms used in this standard.

4.1.1 ACA: Auto contingent allegiance

4.1.2 ACA breakthrough command: A tagged command that has been issued with the tag field set to ACA BREAKTHROUGH. Such commands are legal if and only if an ACA condition is present on the target LUN/TRN.

4.1.3 AEN: Asynchronous event notification.

4.1.4 Application: The source of I/O commands and I/O control commands.

4.1.5 Auto contingent allegiance: The state of a target LUN following the return of a CHECK CONDITION status, after which further I/O processing is suspended except for ACA breakthrough commands.

4.1.6 byte: In this standard, this term indicates an 8-bit construct.

4.1.7 Command descriptor block: The structure used to communicate commands from an initiator to a target.

4.1.8 Common Exchange Block: A structure that is used to pass commands and data over a packetized SCSI interconnect. The format of an exchange block is the same for all SCSI implementations that support the packetized SCSI protocol.

4.1.9 Current I/O Process: An I/O process that is sending or receiving information over the physical interconnect.

4.1.10 Device Model: The object, within a target LUN, that performs the set of device operations defined by one of the SCSI device specifications (SBC, SBS or SCCS).

4.1.11 Domain: The set of SCSI devices and physical interconnects that, collectively, comprise a single, fully-connected network. In such a network, all devices have the same view of the device configuration and each device may communicate with any other device in the domain. A device may belong to multiple domains.

4.1.12 function: An interface between objects residing on the same SCSI device.

4.1.13 initiator: An SCSI device (usually a host system) that requests an I/O process to be performed by another SCSI device (a target).

4.1.14 initiator I/O Process: An I/O process, residing in the initiator, which interacts with a cooperating target I/O process to perform an SCSI command. An initiator I/O process is created when an application command is received and is normally terminated following the receipt of COMMAND COMPLETE status message from the target or the transmission of a RELEASE RECOVERY I/O control command. An initiator I/O Process also ends in the event of a fatal interconnect exception or whenever an ABORT, ABORT TAG, BUS DEVICE RESET, or CLEAR QUEUE I/O control command is sent.

4.1.15 interconnect: A physical pathway for the transfer of commands and data between SCSI devices in a domain.

4.1.16 interconnect exception: An interconnect error resulting in the failure to deliver one or more packets comprising a message.

4.1.17 I/O Process Control Object: An object residing in the initiator or target that executes incoming process control commands or services outgoing commands by interacting with a cooperating I/O process control object on another SCSI device.

4.1.18 I/O process: An object residing in the initiator or target LUN/TRN that executes an SCSI command or series of linked commands. Each SCSI command or series of linked commands causes the creation of a pair of cooperating I/O processes, within the initiator and target, which interact to execute the operation.

4.1.19 I/O Process Queue: A target-resident list of uncompleted I/O processes.

4.1.20 logical interconnect: One or more physical interconnects that, together, constitute the data path for a domain. When a logical interconnect is made up of multiple physical interconnects, some combination of software and hardware is used to make the data path appear monolithic to higher layers.

4.1.21 logical unit: A physical or virtual peripheral device addressable through a target.

4.1.22 logical unit number: An encoded xxx-bit identifier for the logical unit.

4.1.23 LUN: Logical unit number.

4.1.24 mandatory: The referenced item is required to claim compliance with this standard

4.1.25 multipath interconnect: An interconnect comprised of several independent physical paths between the SCSI devices in a domain. These paths may be used to increase bandwidth, through "striping", and to improve availability through redundancy. In an SCSI environment, multipath interconnect management may performed by the transport service in a manner that is transparent to the device.

4.1.26 nexus: A relationship between cooperating initiator and target I/O processes that begins when the command descriptor block is sent and ends with termination of one or both processes.

4.1.27 Object: An architectural abstraction that encapsulates data types, functions, or other objects.

4.1.28 optional: The referenced item is not required to claim compliance with this standard.

4.1.29 packetized SCSI: An SCSI variant that uses a packetized interconnect as the transport mechanism for commands and data.

4.1.30 port: A single attachment to a physical interconnect from an SCSI device. SCSI devices may have multiple ports, each attached to the same or a different physical interconnect.

4.1.31 port address: The physical address of a port attached to an interconnect.

4.1.32 protocol: The rules governing the content and exchange of messages passed between cooperating objects residing on different SCSI devices in a domain.

4.1.33 reserved: The term used for bits, fields, signals, and code values that are set aside for future standardization.

4.1.34 SCSI: Either SCSI-2 or SCSI-3.

4.1.35 SCSI-2: The Small Computer System Interface - 2 (X3.131-199X).

4.1.36 SCSI Device: A physical device in an SCSI domain.

4.1.37 SCSI device address: An address by which an SCSI device is referenced within a domain. The device address need not correspond to the port address. Each SCSI device within a domain may have one and only one SCSI device address.

4.1.38 SCSI interlocked protocol: A protocol, defined by the SIP standard, that uses low-level bus signals to control and synchronize the states of the initiator and target. In the SIP protocol, such signals are used to initiate and transfer data associated with SCSI control, command and device I/O operations.

4.1.39 SCSI packetized protocol: A protocol, defined in this standard, that is designed to be implemented using any packetized interconnect technology. In a packetized protocol, message transactions replace the use of low-level bus signals as a way of initiating and passing data associated with control, command and device I/O operations. Except to control the transfer of data, bus signals play no part in an I/O transaction that uses the packetized protocol.

4.1.40 target: An SCSI device that performs an I/O operation requested by an initiator.

4.1.41 target I/O Process: An I/O process, residing in the target device, that interacts with a cooperating initiator I/O process to perform an SCSI command. A target I/O process is created when an application command is received from the initiator and normally terminates by sending a COMMAND COMPLETE status message or when a RELEASE RECOVERY I/O control command is received. A target I/O Process also ends in the event of a message delivery failure, hard reset, or whenever an ABORT, ABORT TAG, BUS DEVICE RESET, or CLEAR QUEUE I/O Control command is sent or received.

4.1.42 third-party command: An SCSI command, such as COPY, which requires the target device to assume the initiator role and transmit one or more commands to another device on behalf of the original initiator.

4.1.43 transport control service: A combination of hardware and software that controls interconnect access. In a packetized SCSI device, the transport service transmits and receives common exchange blocks over the physical interconnect. When transmitting a block, this service maps the destination SCSI address into the address of the associated physical port, decomposes blocks into packets as required by the physical interconnect and transmits each packet to the destination port. When receiving a block, the

transport service is responsible for the reassembly of message packets into exchange blocks and the delivery of exchange blocks in the order in which they were sent by the originator.

When the device is attached to a multipath interconnect, the transport service manages the flow of traffic over the interconnect in a way that is transparent to the SCSI device.

4.1.44 transport-initiated reset: Hard reset initiated by the transport control services layer. This function is provided as a way to externally reset a device to it's power-on state when an error is detected that prevents the device from responding to I/O control commands.

4.1.45 word: In this standard, this term indicates a 1-byte, 2-byte, or 4-byte construct.

4.1.46 xx: Digits 0-9, except those used as section numbers, in the text of this standard that are not immediately followed by lower-case "b" or "h" are decimal values. Large Numbers are not separated by commas or spaces (e.g., 12345; not 12,345 or 12 345).

4.1.47 xxb: Digits 0 and 1 immediately followed by lower-case "b" are binary values.

4.1.48 xxh: Digits 0-9 and the upper-case letters "A"-"F" immediately followed by lower-case "h" are hexadecimal values.

4.2 Object Notation

The following notational conventions are used to describe composite objects:

+		"together with" as in A = B + C. Object A contains both B and C.
[|]		"select one of" as in A = [B|C]. Object A contains either B or C, but not both. This is equivalent to an "exclusive or" operation.
()		"optional" as in A = (B) + (C + D). Object A contains either B or C + D. This is equivalent to an "inclusive or" operation.
{ }	"iterations of"	A set of objects enclosed within curly brackets may occur any number of times in a given instance. The brackets may be indexed: For example, M{}N indicates any number of instances from M to N. Thus: {...}3 implies 0, 1, 2 or 3 instances 3{...} implies 3 of more instances 3{...}3 implies exactly 3 iterations
" "	literal	Symbols enclosed within quotes literally constitute the object.

4.3 Editorial Conventions

Certain words and terms used in this standard have a specific meaning beyond the normal English meaning. These words and terms are defined either in the glossary or in the text where they first appear. Names of signals and phases are in all uppercase. Lower case is used for words having the normal English meaning.

5 Overview of the Architecture

[ED. Note - this section will be generic to SPP and SIP]

The following elements are described in this section:

a. SCSI domain - A logical interconnect and the associated device configuration that is capable of performing all SCSI I/O operations.

b. Logical interconnect - The set of physical interconnects that are used for the transmission of commands and data between the devices in a domain. Multiple physical interconnects may be used to increase availability and throughput.

c. SCSI device - A physical device that implements one of the device models described in the SBC, SSC or SCC standards.

5.1 SCSI Architectural Elements

The SCSI architecture is described in terms of objects, functions and protocols.

The SCSI objects defined in this standard are abstractions that encapsulate a set of operations, data types and other objects that are related in some way.

A function is an interface between objects that reside in the same SCSI device.

A protocol is an interface between objects on different SCSI devices. Protocols are described by a set of rules governing the content and exchange of messages passed between cooperating objects.

5.2 SCSI Domain

An SCSI domain is a set of SCSI devices and interconnects that, together, comprise the fully connected network shown in figure ???. In this topology, every device in the domain may communicate with all other devices and each device sees the same configuration. These properties allow a third-party I/O operation to be requested between any two SCSI devices in the domain.

5.3 Logical Interconnect

A logical interconnect is made up of one or more physical interconnects that are used for the transfer of commands and data between devices in an SCSI domain. A logical interconnect may have several redundant physical interconnects that are used to enhance data throughput or system availability. As described in ????, one function of the Transport Services implementation within an SCSI device is to hide such aspects of the interconnect from other parts of the device-resident subsystem.

One example of a multi-path system might be a collection of dual-ported SCSI devices, with separate physical interconnects connected to each port for increased availability. In this configuration, the Transport Services implementation within an initiator could dynamically select a rail for arbitration based on whether or not the associated physical interconnect was in use. As described in ???, a bus reset or other failure on one rail would not effect data transfers occurring on the other.

A logical interconnect may also be comprised of mixed interconnect technologies, such as short and long-haul physical interconnects, that are interfaced via bridges or other devices such that a seamless view of the data path is presented to all devices in the domain.

The formal definition of a logical interconnect is:

Logical_Interconnect = 1 { *Physical_Interconnect* }

5.4 Common Exchange Block

[Ed. note: This will be a detailed description of the CEB, including the formats of embedded fields, such as SCSI Device Address, LUN ... etc.]

5.5 SCSI Device

An SCSI device is a physical element containing the following objects:

a. One or more SCSI ports,

b. A transport control services object,

c. An initiator and zero or more target LUN/TRN objects or,

d. An optional initiator and one or more target LUN/TRN objects.

Description:

SCSI_device = 1{*SCSI_port* + *Transport_Services* +
[*Initiator* + 0{*Target_LUN|TRN* } | 1{*Target_LUN|TRN*}]

5.5.1 Transport Control Services

Transport control services is a facility for communication between objects which conceals the details of the ports and physical interconnects from the rest of the I/O subsystem. This object performs the following functions:

For outgoing messages:

a. Decomposes the common exchange block into packets as required by the physical interconnect,

b. In a multi-port configuration, selects the port to be used.

c. Maps the receiver's SCSI device address to a physical port address.

d. Initiates packet transfer over the SCSI port.

For incoming messages, this service:

a. Accepts packets received via one or more SCSI ports and reassembles them into common exchange blocks,

b. Delivers common exchange blocks to the receiving object, preserving the order in which the blocks were received.

In addition to the above, this service notifies the device whenever a fatal interconnect error is detected.

5.5.2 Port

A port is the point of attachment to a physical interconnect and has an implementation-specific physical address that is referenced by other connected ports.

In packetized SCSI, the port's function is the sequenced exchange of packets with another port without loss, error or duplication. On transmission, the port converts packets into bit streams for transmission over the interconnect, adding any required error detection information. When receiving data, the port validates the incoming bit stream, performs error recovery and converts the bit stream to packets in memory, maintaining the sequence in which packets were received from the sender.

5.5.3 Initiator

An initiator object originates requests for I/O services by sending commands to SCSI target devices. An initiator is composed of the following:

a. Initiator I/O process - Interacts with a cooperating target I/O process to execute a command or series of linked commands. An initiator I/O process consists of the procedures for servicing an I/O request, a data buffer, if required by the command, and an optional buffer for sense data.

b. Initiator I/O Process control object - Interacts with a target I/O process control object to create or terminate I/O processes and return process completion status to the application.

Description:

Initiator = 0{Initiator_I/O_process} + Initiator_I/O_process_control

Initiator_I/O_process = Data_buffer+{Sense_buffer}

5.5.4 Target LUN/TRN

The target LUN/TRN responds to initiator I/O requests and consists of the following objects:

a. Target I/O process control object - Creates, deletes and queues processes as requested by the initiator,

b. I/O process queue - List of uncompleted target I/O Processes

c. Device model - Object that implements one of the device models defined in the SBC, SBS or SCCS standards.

Description:

Target_LUN/TRN = Target_I/O_process_control + I/O_Process_queue +
{ Target_I/O_Process }+Device_model

5.6 Transport service requirements

[Ed. Note. This section discusses the generic requirements for the implementation-specific transport services layer, including required initiator and target behavior in the event of packet delivery errors. The following additional contingencies will be covered:

a. Hard Reset/Hot/warm swap detection in a packetized environment.

b. Transport-initiated reset - escape mechanism in the event that a device is believed to be insane. Ie: there is some brain damage that prevents the device from servicing normal I/O control commands, such as BUS DEVICE RESET. This would the packetized analog of an SIP RESET signal.]

5.7 Common Exchange Block Protocol

[ed. Note: This section defines the protocol for interchanging CEBs.]

6 I/O Control Operations

(Ed. Note - See SCSI-2, section 6.]

6.1 SCSI Commands and Status

[ed. note - see SCSI-2, section 6.1]

6.2 Command Descriptor Block

[ed. note - see SCSI-2, section 6.2, for starters]

6.3 Status

[ed. note - Based on SCSI-2, section 6.3]

6.4 Examples

[e. note - new packetized examples, including linked commands]

6.5 Command Processing Considerations and Exception Conditions

[ed. note - Based on SCSI-2, extensively revised for SCSI-3.]

6.6 Contingent Allegiance

[ed. note - will this be defined for SCSI-3?]

6.7 Extended Contingent Allegiance Condition

[ed. note - will this be defined for SCSI-3?]

6.8 Queued I/O Processes

[Ed. note - SCSI-3 Queuing Model goes here]

6.9 Unit Attention Condition

[ed. note - As modified for SCSI-3]

6.10 Multiport SCSI Devices

[ed. note - specifies the behavior of multi-port devices.]