Date: Oct 10, 2000

To: T10 Committee (SCSI)

From: George Penokie (IBM)

Subject: working draft SCSI-3 Controller Commands

1 Overview

Do to the negative vote in the 9/14/00 plenary on proposal 00-246r0 some corrections to SPI-4, not related to the multiple masks issue, could not be placed into SPI-4 rev 1. Those corrections were needed to prevent possible incorrect implantations. Normally in that would be handled at the next meeting, however, there have my requests for a new version of SPI-4 because of all the other changes that were approved. As result I am placing an 'editor's version' of SPI-4 rev 1 into this proposal. The changes between rev 0 and this rev are contained in the revisions section below and marked within the document.

Working Draft American National Standard

T10 Project 1365D

Revision ~~0~~1
~~28~~05-~~May~~Oct-2000

Information technology - SCSI Parallel Interface-4 (SPI-4)

T10 Technical Editor: George O. Penokie
~~IBM~~Tivoli Systems
MS ~~Z9V~~2C6
3605 Highway 52 N.
Rochester, MN 55901
USA

Telephone: 507-253-5208
Facsimile: 507-253-2880
Email: gpenokie@tivoli.com

Reference number
ISO/IEC ****** : 200x
ANSI NCITS. - 200x

Printed Monday, October 16, 2000 3:29 pm

POINTS OF CONTACT:

T10 Chair
John B. Lohmeyer
LSI
4420 ArrowsWest Drive
Colo Spgs, CO 80907-3444
Tel: (719) 533-7560
Fax: (719) 593-7183
Email: lohmeyer@t10.org

T10 Vice-Chair
George O. Penokie
~~IBM~~ Tivoli Systems
3605 Highway 52 N. MS ~~Z9V~~2C6
Rochester, MN 55901
Tel: (507) 253-5208
Fax: (507) 253-2880
Email: gpenokie@tivoli.com

NCITS Secretariat
NCITS Secretariat
1250 Eye Street, NW Suite 200
Washington, DC 20005
Telephone: 202-737-8888
Facsimile: 202-638-4922
Email: ncits@itic.org

T10 Reflector Internet address for subscription to the T10 reflector: majordomo@t10.org
The message body should be: subscribe t10
Internet address for distribution via T10 reflector: T10@t10.org
Internet address to unsubscribe from the T10 reflector: majordomo@t10.org
The message body should be: unsubscribe t10

Document Distribution
Global Engineering
15 Inverness Way East
Englewood, CO 80112-5704
Telephone: 303-792-2181 or
800-854-7179
Facsimile: 303-792-2192

American National Standard
for Information Technology

SCSI Parallel Interface-4 (SPI-4)

Secretariat
Information Technology Industry Council

Approved mm.dd.yy

American National Standards Institute, Inc.

ABSTRACT

This standard defines mechanical, electrical, timing requirements, command, and the task management delivery protocol requirements to transfer commands and data between SCSI devices attached to a SCSI parallel interface. This standard is intended to be used in conjunction with the SCSI command sets. The resulting interface facilitates the interconnection of computers and intelligent peripherals and thus provides a common interface standard for both system integrators and suppliers of intelligent peripherals.

American National Standard

Approval of an American National Standard requires verification by ANSI that the requirements for due process, consensus, and other criteria for approval have been met by the standards developer. Consensus is established when, in the judgment of the ANSI Board of Standards Review, substantial agreement has been reached by directly and materially affected interests. Substantial agreement means much more than a simple majority, but not necessarily unanimity. Consensus requires that all views and objections be considered, and that effort be made towards their resolution.

The use of American National Standards is completely voluntary; their existence does not in any respect preclude anyone, whether he has approved the standards or not, from manufacturing, marketing, purchasing, or using products, processes, or procedures not conforming to the standards.

The American National Standards Institute does not develop standards and will in no circumstances give interpretation on any American National Standard. Moreover, no person shall have the right or authority to issue an interpretation of an American National Standard in the name of the American National Standards Institute. Requests for interpretations should be addressed to the secretariat or sponsor whose name appears on the title page of this standard.

CAUTION NOTICE: This American National Standard may be revised or withdrawn at any time. The procedures of the American National Standards Institute require that action be taken periodically to reaffirm, revise, or withdraw this standard. Purchasers of American National Standards may receive current information on all standards by calling or writing the American National Standards Institute.

CAUTION: The developers of this standard have requested that holders of patents that may be required for the implementation of the standard, 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 standard. As of the date of publication of this standard, following calls for the identification of patents that may be required for the implementation of the standard, no such claims ~~has been~~ have been made. No further patent search is conducted by the developer or the publisher in respect to any standard it processes. No representation is made or implied that licenses are not required to avoid infringement in the use of this standard.

Published by
American National Standards Institute
11 W. 42nd Street, New York, New York 10036

Printer in the United States of America

Contents

Tables

Figures

Revision History

Revision 0

a) Made changes and additions per the Fast-160 (99-295r5) proposal.
b) Made changes and additions per the SPI-4 training pattern DC signal strengths (00-231r0) proposal.
c) Made data group transfers prohibited on paced data transfers per plenary vote.
d) Changed weak driver level in to 60% - 78% per plenary vote.
e) Changed receiver threshold, driver asymmetry, and minimum drive level numbers to TBD per plenary vote.
f) Added in an annex on Expander Topics (00-199r2).
g) Added in editors notes as where changes will be required when the Flow Control (00-142r4) proposal is made available.
h) 99-145r9 is not available and it is not clear from r8 what changes are supposed to be made to SPI-4 relating to this proposal.
i) Made changes relating to 2/11/2000 email message on SPI-3 from QLogic. It requested it be made clear that a bus free occurs anytime IU transfers are enabled or disabled.

Revision 1

a) Made correction relating to 6/6/2000 email message from IBM. There are two places were negating should have been asserting. One in section 10.8.4.2.2 and another in section 10.8.4.2.3.
b) Made changes relating to 5/31/2000 email message from TI. The following was added to Annex A section A.2.6: For paced transfers the output signal rise or fall times (see t_r in figure A.8) between 0,2 and 0,8 of V_{SS} shall be greater than or equal to 1 ns and less than or equal to 2,5 ns. In addition a new figure was added to section 4.8 for paced transfers.
c) The following was removed from annex F as there are no bridging expanders in that annex: Bridging expander: Devices that couple a bus segment to another SCSI segment or another kind of port by using addressable SCSI ports in the device. Bridging expanders have SCSI IDs on all ports, participate in SCSI arbitration and messaging, and are “devices” in the SCSI sense. Bridging expanders are not defined in this standard. This change was as a result of the July 2000 SPI-4 working group meeting.
d) The following was removed from section F.4 item c as it contained no useful information: 'shall be sent by the simple expander, however, messages sent from initiators and targets could be read (e.g., the simple expander could require the negotiated data phase speed or some other variable property of a transaction of an initiator/target connection'. This change was as a result of the July 2000 SPI-4 working group meeting.
e) The following was removed from section F.8.5.2 as it contained no useful information: 'Care shall be exercised when considering expanders to understand the capabilities of the expanders being used.' This change was as a result of the July 2000 SPI-4 working group meeting.
f) The term 'phase change' was being misused in several places in the document. It was replaced with wording along the lines of 'a change to the C/D, I/O, or MSG signals'. This change was as a result of the July 2000 SPI-4 working group meeting.
g) Placed flow control and read streaming from accepted proposals 00-142r4 and 00-285r1.
h) Placed the saving of training values option into PPR and the training section from the accepted proposal 00-270r0.
i) Per request from the July 2000 SPI-4 working group meeting: Changed the following sentence from the PPR IU_REQ paragraph: 'If the IU_REQ bit is changed from the previous agreement (i.e., zero to one or one to zero) as a result of a negotiation the target shall go to a BUS FREE phase on completion of the negotiation.' to 'Each time the IU_REQ bit is changed from the previous agreement (i.e., zero to one or one to zero) as a result of a negotiation the target shall go to a BUS FREE phase on completion of the negotiation.'
j) Replaced the TBD in row 3 of table A.1 with 100 mv and added in a new figure that shows the Vn vs Va for f-160 transfers and a new row in table A.2 that has the equations for Vn vs Va for f-160 transfers.
k) Fixed the description of the P1 signal as it did not include it's operation during paced transfers.

l) Change section 10.13 to allow the SEL signal to be changed between phases as required to indicate the start of the training pattern.
m) Removed the pace_on function from the PPR message per 00-292r2.
n) The following proposals have been added:
    a) 00-257r3 - Expander Communication Protocol
    b) 00-311r1 - Proposal for an “assertion handshaking” protocol for Ultra320 SCSI in SPI-4
    c) 00-321r1 - Disabling Precompensation
    d) 00-322r2 - SPI-4 Clarifications
    e) 00-323r4 - Proposal for revision of the timing tables and definitions on SPI-4
    f) 00-324r3 - Getting Training Started
    g) 00-325r0 - Vendor Unique IU type
    h) 00-326r1 - Packetized streaming clarification
    i) 00-333r0 - Standard is a functional description
    j) 00-335r1 - SPI-4 IU Handling Clean-up
    k) 00-343r2 - Timing for ATN and P_CRCA in SPI-4
    l) 00-353r1 - Proposal for revised setup and hold figures to be included in SPI-4
o) Revised section 10.8.4.4 Paced information unit transfer as recommended in the 9/12/00 SPI-4 working group meeting
p) Added in the terminator leakage which was placed into document 00-246r7 from 00-320r0 and was agreed to in the 9/12/00 SPI-4 working group meeting.
q) Added in the corrected driver equations in section A.2.2 Differential output voltage, V_S which was placed into document 00-246r7 from 00-319r0 and 00-347r0 and was agreed to in the 9/12/00 SPI-4 working group meeting.
r) The mask for isolated nonprecompensated signals and for clocked and non-isolated signals have been added from 00-347r0. The precompensated mask were not added as they were defeated by plenary vote (9/14/00) when proposal 00-246r7 was not accepted.

Foreword (This foreword is not part of this standard)

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

This standard was processed and approved for submittal to ANSI by National Committee for Information Technology Standards (NCITS). Committee approval of this standard does not necessarily imply that all committee members voted for approval. At the time it approved this standard, NCITS had the following members:

Karen Higginbottom, Chair
David Michael, Vice-chair
Monica Vago, Secretary

(NCITS Membership to be inserted)

Technical Committee T10 on Lower Level Interfaces, that developed this standard, had the following members:

John B. Lohmeyer, Chair
George O. Penokie, Vice-Chair
Ralph O. Weber, Secretary

(To be added in at start of first public review)

Introduction

The SCSI protocol is designed to provide an efficient peer-to-peer I/O bus with the maximum number of hosts and peripherals determined by the bus width (8 or 16). Data may be transferred asynchronously or synchronously at rates that depend primarily on device implementation and cable length.

The SCSI Parallel Interface-3 standard is divided into the following clauses:

- Clause 1 is the scope;
- Clause 2 enumerates the normative references that apply to this standard;
- Clause 3 describes the definitions, symbols, conventions and abbreviations used in this standard;
- Clause 4 describes the SCSI parallel interface model used in this standard;
- Clause 5 describes the connectors;
- Clause 6 describes the cable characteristics;
- Clause 7 describes the electrical characteristics;
- Clause 8 describes the SCSI bus signals;
- Clause 9 describes the SCSI parallel bus timing;
- Clause 10 describes the SCSI bus phases;
- Clause 11 describes the DATA BUS protection;
- Clause 12 describes the SCSI bus conditions;
- Clause 13 describes the SCSI bus phase sequences;
- Clause 14 describes the SPI information unit sequences;
- Clause 15 describes the SCSI pointers;
- Clause 16 describes the SCSI messages;
- Clause 17 describes the Command processing considerations and exception conditions;
- Clause 18 describes the SCSI management features for the SCSI parallel interface;
- Clause 19 describes the SCSI parallel interface services;

Annexes A, B, C, D, E, F, and ~~F~~ G form an integral part of this standard. Annexes ~~G~~ H through ~~N~~ O are for information purposes only.

American National Standard
for Information Technology -

SCSI Parallel Interface-4 (SPI-4)

1 Scope

This standard defines the mechanical, electrical, timing, and protocol requirements of the SCSI parallel interface to allow conforming SCSI devices to inter-operate. The SCSI parallel interface is a local I/O bus that may be operated over a wide range of transfer rates. The objectives of the SCSI parallel interface are:

a) To provide host computers with device independence within a class of devices. Thus, different disk drives, tape drives, printers, optical media drives, and other SCSI devices may be added to the host computers without requiring modifications to generic system hardware. Provision is made for the addition of special features and functions through the use of vendor-specific options. Reserved areas are provided for future standardization.
b) To provide compatibility such that conforming SCSI-2, SPI-2, and SPI-3 devices may interoperate with SPI-4 devices given that the systems engineering is correctly done. Conforming SCSI-2, SPI-2, and SPI-3 devices should respond in an acceptable manner to reject SPI-4 protocol extensions. SPI-4 protocol extensions are designed to be permissive of such rejections and thus allow SCSI-2, SPI-2, and SPI-3 devices to continue operation without requiring the use of the extensions.

The interface protocol includes provision for the connection of multiple initiators (SCSI devices capable of initiating an I/O process) and multiple targets (SCSI devices capable of responding to a request to perform an I/O process). Distributed arbitration (i.e., bus-contention logic) is built into the architecture of this standard. A default priority system awards interface control to the highest priority SCSI device that is contending for use of the bus and an optional fairness algorithm is defined.

This standard defines the physical attributes of an input/output bus for interconnecting computers and peripheral devices.

The set of SCSI standards specifies the interfaces, functions, and operations necessary to ensure interoperability between conforming SCSI implementations. This standard is a functional description. Conforming implementations may employ any design technique that does not violate interoperability.

This standard has made obsolete the following:

a)

Figure 1 is intended to show the general structure of SCSI standards. The figure is not intended to imply a relationship such as a hierarchy, protocol stack, or system architecture.

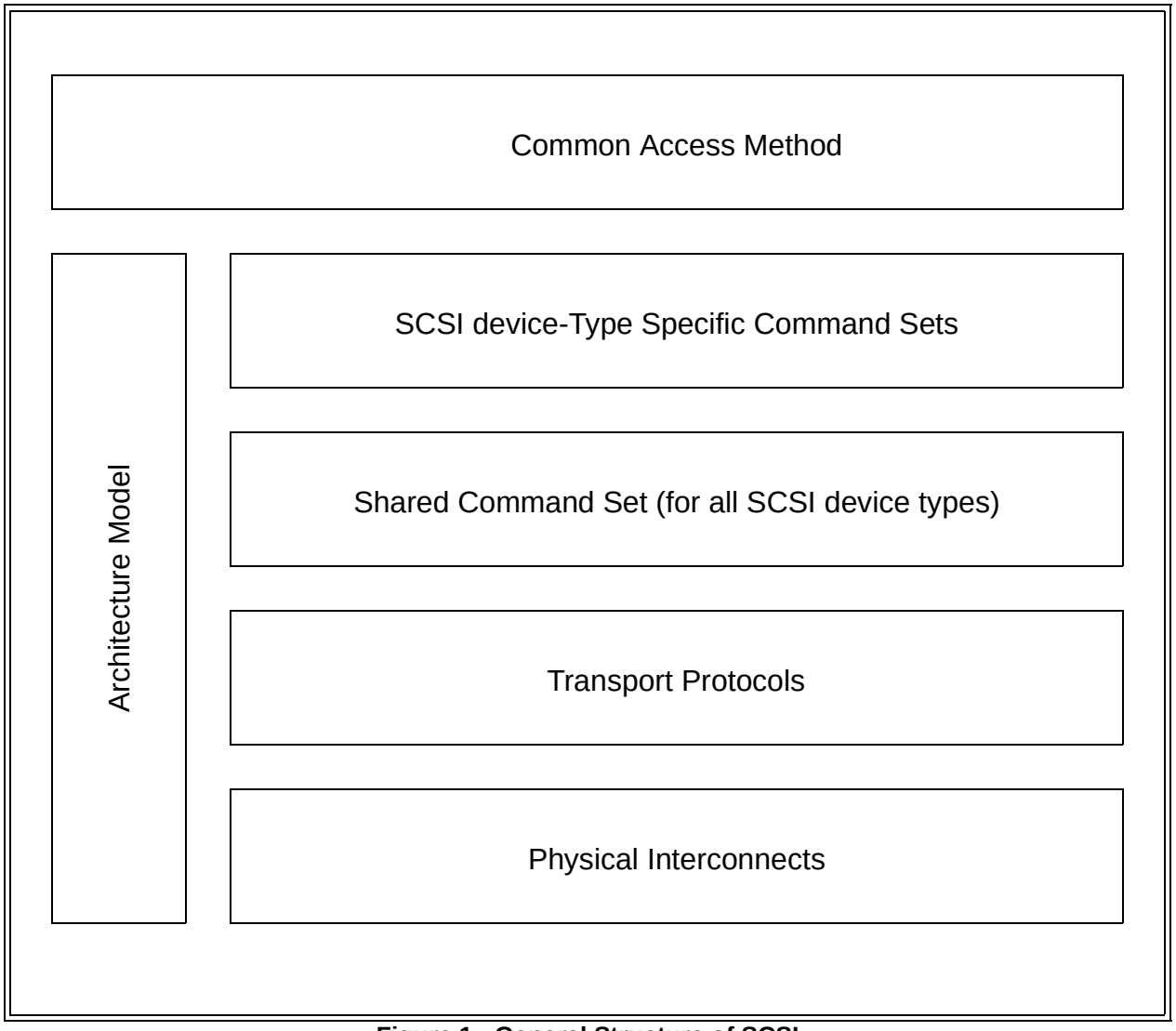


Figure 1 - General Structure of SCSI

At the time this standard was generated examples of the SCSI general structure included:

Physical Interconnects:

AT Attachment with Packet Interface Extension [NCITS.317-1998]
Fibre Channel Arbitrated Loop-2 [T11/1133D]
Fibre Channel - Physical and Signaling Interface [X3.230-1994]
High Performance Serial Bus [IEEE 1394-1995]
SCSI Parallel Interface - 2 [X3.302-1998]
SCSI Parallel Interface - 3 [NCITS.336-200x]
SCSI Parallel Interface - 4 [This standard]
Serial Storage Architecture Physical Layer 1 [X3.293-1996]
Serial Storage Architecture Physical Layer 2 [NCITS.307-1998]

Transport Protocols:

SCSI Parallel Interface - 2 [X3.302-1998]
Serial Storage Architecture Transport Layer 1 [X3.295-1996]
SCSI-3 Fibre Channel Protocol [X3.269-1996]

SCSI Fibre Channel Protocol - 2 [T10/1144D]
SCSI Serial Bus Protocol - 2 [NCITS.325-1998]
Serial Storage Architecture SCSI-2 Protocol [X3.294-1996]
Serial Storage Architecture SCSI-3 Protocol [NCITS.309-1998]
Serial Storage Architecture Transport Layer 2 [NCITS.308-1998]

Shared Command Set:

SCSI-3 Primary Commands standard [X3.301-1997]
SCSI Primary Commands-2 standard [T10/1236D]

Device-Type Specific Commands Sets:

SCSI-3 Block Commands [NCITS.306-1998]
SCSI-3 Enclosure Services [NCITS.305-1998]
SCSI-3 Stream Commands [T10/997D]
SCSI-3 Medium Changer Commands [T10/999D]
SCSI-3 Controller Commands [X3.276-1997]
SCSI Controller Commands - 2 [T10/1225D]
SCSI-3 Multimedia Command Set [X3.304-1997]
SCSI Multimedia Command Set - 2 [T10/1228D]

Architecture Model:

SCSI-3 Architecture Model [X3.270-1996]
SCSI Architecture Model - 2 [T10/1157D]

Common Access Method:

SCSI Common Access Method [X3.232-1996]

The term SCSI is used wherever it is not necessary to distinguish between the versions of SCSI. The Small Computer System Interface - 2 (ANSI X3.131-1994) is referred to herein as SCSI-2.

2 Normative references

2.1 Normative references

The following standards contain provisions that, by reference in the text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent editions of the standards listed below.

Copies of the following documents may be obtained from ANSI: approved ANSI standards, approved and draft international and regional standards (ISO, IEC, CEN/CENELEC, ITUT), and approved and draft foreign standards (including BSI, JIS, and DIN). For further information, contact ANSI Customer Service Department at 212-642-4900 (phone), 212-302-1286 (fax) or via the World Wide Web at http://www.ansi.org.

Additional availability contact information is provided below as needed.

2.2 Approved references

ISO/IEC 8482:1993-12, Information technology - Telecommunications and information exchange between systems - Twisted pair multipoint interconnections

EIA-700A0AE (SP-3651), Detail Specification for Trapezoidal Connectors with Non-removable Ribbon Contacts on 1.27 mm Pitch Double Row used with Single Connector Attachments (SCA-2)

EIA-700A0AF (SP-3652), Detail Specification for Trapezoidal Connector 0.8 mm Pitch used with Very High Density Cable Interconnect (VHDCI)

IEC 60512-2:1985-11, Low Level Contact Resistance Test Procedure for Electric Connectors

IEC 60512-11-7:1996-01, Standard Practice For Conducting Mixed Flowing Gas Environmental Tests

~~*ANSI Y14.5M:1994*, Dimensioning and Tolerancing~~

ISO 129 , Technical Drawings - Dimensioning - General Principals

ISO 1660, Technical Drawings - Dimensioning And Tolerancing

ISO/IEC 14776-112, SCSI Parallel Interface-2 standard

2.3 References under development

At the time of publication, the following referenced standards were still under development. For information on the current status of the document, or regarding availability, contact the relevant standards body or other organization as indicated.

ISO/IEC 14776-412, SCSI Architecture Model-2 standard

ISO/IEC 14776-113, SCSI Parallel Interface-3 standard

T10/1236D, SCSI Primary Commands-2 standard

NOTE 1 - For more information on the current status of the document, contact the NTCIS Secretariat at 202-737-8888 (phone), 202-638-4922 (fax) or via Email at nctis@itic.org. To obtain copies of this document, contact Global Engineering at 15 Inverness Way, East Englewood, CO 80112-5704 at 303-792-2181 (phone), 800-854-7179 (phone), or 303-792-2192 (fax).

2.4 Other references

For information on the current status of the listed document(s), or regarding availability, contact the indicated organization.

SFF-8451, SCA-2 Unshielded Connections

NOTE 2 - For more information on the current status of the document, contact the SFF committee at 408-867-6630 (phone), or 408-867-2115 (fax). To obtain copies of this document, contact the SFF committee at 14426 Black Walnut Court, Saratoga, CA 95070 at 408-867-6630 (phone) or 408-741-1600 (fax).

ASTM D-4566, Standard Test Methods for Electrical Performance Properties of Insulations and Jackets for Telecommunications Wire and Cable
ASTM B827, Standard Practice For Conducting Mixed Flowing Gas Environmental Tests
IEEE 1364, Verilog® Hardware Description Language

3 Definitions, symbols, abbreviations, and conventions

3.1 Definitions

3.1.1 A cable: A 50-conductor cable that provides an 8-bit DATA BUS and control signals.

3.1.2 agent: Carries out the actions of a requested service following the rules of the protocol.

3.1.3 application client: An object that is the source of SCSI commands. Further definition of an application client is found in the SCSI Architecture Model-2 standard.

3.1.4 asymmetry: The total (peak to peak) time difference of a SCSI data bus signal from the nominal time of the signal at a given point (e.g., the difference in time of a one versus a zero in a continuous data pattern of alternating ones and zeros).

3.1.5 asynchronous event notification: An optional procedure used by targets to notify initiators of events that occur when a pending task does not exist for that initiator.

3.1.6 asynchronous transfer: An information transfer that uses the asynchronous REQ/ACK handshake with an offset of zero.

3.1.7 auto-contingent allegiance: An optional condition of a task set following the return of a CHECK CONDITION status. See the SCSI Architecture Model-2 standard for a detailed definition of auto-contingent allegiance.

3.1.8 bus path: The electrical path directly between the bus terminators.

3.1.9 byte: Indicates an 8-bit construct.

3.1.10 confirmation: The last step of a confirmed service informing the upper protocol layer that the requested service has been completed.

3.1.11 confirmed service: A service available at the protocol service interface, that requires confirmation of completion. The confirmed service consists of the request and confirmation steps and optionally the indication and response steps.

3.1.12 contact: The electrically-conductive portion of a connector associated with a single conductor in a cable.

3.1.13 contingent allegiance: An optional condition of a task set following the return of a CHECK CONDITION status. A detailed definition of contingent allegiance may be found in the SCSI Architecture Model-2 standard.

3.1.14 cyclic redundancy check (CRC): An error detecting code used to detect the validity of data.

3.1.15 current task: A task that is in the process of sending messages, status, transferring data, or transferring command data to or from the initiator.

3.1.16 DATA BUS: An 8-bit or 16-bit bus (see 8.2).

3.1.17 data field: The portion of a data group that contains data bytes.

3.1.18 data group: A sequence of data bytes, any pad bytes, and the four pCRC bytes during a DT DATA IN phase or a DT DATA OUT phase.

3.1.19 data group transfer: Parallel transfers that transfer data and pCRC information using only data groups.

3.1.20 device server: An object within the logical unit that executes SCSI tasks according to the rules for task management as described in the SCSI Architecture Model-2 standard.

3.1.21 differential: A signaling alternative that uses drivers and receivers with two complementary signals to improve signal-to-noise ratios and increase maximum cable lengths (also see 3.1.89 SE).

3.1.22 double transition (DT): The latching of data on both the assertion edge and the negation edge of the REQ or ACK signals.

3.1.23 driver: The circuitry used to control the state of the bus.

3.1.24 exception condition: Any event that causes a SCSI device to enter an auto-contingent allegiance or contingent allegiance condition.

3.1.25 expander: A device that connects SCSI buses together to form a single SCSI domain.

3.1.26 fast-5: Negotiated to receive synchronous data at a transfer period that translates into a transfer rate of less than or equal to 5 megatransfers per second.

3.1.27 fast-10: Negotiated to receive synchronous data at a transfer period that translates into a transfer rate greater than 5 megatransfers per second and less than or equal to a transfer rate of 10 megatransfers per second.

3.1.28 fast-20: Negotiated to receive synchronous data at a transfer period that translates into a transfer rate greater than 10 megatransfers per second and less than or equal to a transfer rate of 20 megatransfers per second.

3.1.29 fast-40: Negotiated to receive synchronous data at a transfer period that translates into a transfer rate greater than 20 megatransfers per second and less than or equal to a transfer rate of 40 megatransfers per second.

3.1.30 fast-80: Negotiated to receive synchronous data at a transfer period that translates into a transfer rate greater than 40 megatransfers per second and less than or equal to a transfer rate of 80 megatransfers per second.

3.1.31 fast-160: Negotiated to receive synchronous data at a transfer period that translates into a transfer rate of 160 megatransfers per second.

3.1.32 field: A group of one or more contiguous bits.

3.1.33 indication: The second step of a four step confirmed service in reply to a request.

3.1.34 information unit transfer: Parallel transfers that transfer data, status, commands, task attributes, task management, iuCRC, and nexus information using SPI information units.

3.1.35 initial connection: An initial connection is the result of a physical connect. It exists from the assertion of the BSY signal (see 10.6) in a SELECTION phase until the next BUS FREE phase or the next QAS REQUEST message.

3.1.36 initiator: A SCSI device containing application clients that originate device service and task management requests to be processed by a target SCSI device. See the SCSI Architecture Model-2 standard for a detailed definition of an initiator.

3.1.37 I/O process: An I/O process consists of one initial connection or, if information units are enabled, the establishment of a nexus, and zero or more physical or logical reconnections, all pertaining to a single task or a group of tasks. An I/O process begins with the establishment of a nexus. If the SPI information unit transfers are disabled an I/O process normally ends with a COMMAND COMPLETE message. If

information unit transfers are enabled an I/O process normally ends with a SPI L_Q information unit with the type field set to status and the DATA LENGTH field set to zero.

3.1.38 interconnect: The electrical media (including connectors and passive loads) used to connect the TERMPWR, terminators, and SCSI devices in a SCSI bus

3.1.39 I_T nexus: A nexus that exists between an initiator and a target.

3.1.40 I_T_L nexus: A nexus that exists between an initiator, a target, and a logical unit. This relationship replaces the prior I_T nexus.

3.1.41 I_T_L_Q nexus: A nexus between an initiator, a target, a logical unit, and a queue tag following the successful receipt of a queue tag. This relationship replaces the prior I_T nexus or I_T_L nexus.

3.1.42 intersymbol interference (ISI): The effect of other symbols on the symbol currently being received.

3.1.43 iuCRC protection: The use of CRC to detect DT DATA phase data transmission errors during SPI information unit transfers.

3.1.44 logical connect: Establishes an I_T_L_Q nexus using SPI L_Q information units during an initial connection.

3.1.45 logical disconnect: Reduces the current I_T_L_Q nexus to an I_T nexus.

3.1.46 logical reconnect: Reestablishes an I_T_L_Q nexus from an I_T nexus using SPI L_Q information units.

3.1.47 logical unit: An externally addressable entity within a target. See the SCSI Architecture Model-2 standard for a detailed definition of a logical unit.

3.1.48 logical unit number: An identifier for a logical unit.

3.1.49 magnitude: The absolute value of a number or quantity.

3.1.50 maximum transfer rate: The fastest transfer rate supported by a SCSI device.

3.1.51 megatransfers per second: The repetitive rate that data are transferred across the bus. This is equivalent to megabytes per second on an 8-bit wide bus.

3.1.52 message: One or more bytes transferred between an initiator and a target to do link control, perform task management, and to associate task attributes with commands.

3.1.53 multidrop: A characteristic of the SCSI bus that allows SCSI devices to be connected to the SCSI bus without disrupting the electrical path between the terminators (see 4.3).

3.1.54 multimode single-ended (MSE): A signalling alternative for LVD SCSI devices that employs MSE (see 7.4) drivers and receivers to allow operation when SE SCSI devices are present on the bus.

3.1.55 nexus: A relationship between an initiator and a target, logical unit, or queue tag that begins with an initial connection and ends with the completion of the associated I/O process. This relationship is formed as the result of a task.

3.1.56 object: An architectural abstraction that encapsulates data types, services, or other objects that are related in some way.

3.1.57 odd parity: Odd logical parity, where the parity bit is driven and verified to be that value that makes

the number of assertions on the associated data byte plus the parity bit equal to an odd number (1, 3, 5, 7, or 9). See 3.1.64, parity bit. If an even number of asserted bits are detected at the receiver a parity error occurs.

3.1.58 one: A true signal value or a true condition of a variable.

3.1.59 P cable: A 68-conductor cable or an 80-conductor connector that provides the 16-bit DATA BUS and control signals.

3.1.60 paced transfer: Parallel transfers that transfer information using pacing.

3.1.61 pacing: Use of the ACK or REQ signal as a continuously running clock in combination with the P1 signal to indicate when data is valid.

3.1.62 packetized: A method of transferring information using SPI information units (see 4.9.3.3)

3.1.63 pad field: The portion of a data group that contains pad information.

3.1.64 parity bit: A bit associated with a byte that is used to detect the presence of an odd number of asserted bits within the byte. The parity bit is driven such that the number of logical ones in the byte plus the parity bit is odd.

3.1.65 parallel transfer: The transfer of information using information transfer phases.

3.1.66 path: The cable, printed circuit board or other means for providing the conductors and insulators that connect two or more points.

3.1.67 pCRC field: The portion of a data group that contains pCRC information.

3.1.68 pCRC protection: The use of CRC to detect DT DATA phase data transmission errors during parallel transfers.

3.1.69 pending task: A task that is not the current task.

3.1.70 physical connect: The act of establishing an I_T nexus during a connection.

3.1.71 physical disconnection: The action that occurs when a SCSI device releases control of the SCSI bus, allowing it to go to the BUS FREE phase.

3.1.72 physical reconnect: The act of resuming a nexus to continue a task. A target initiates a physical reconnect when conditions are appropriate for the physical bus to transfer data associated with a nexus between an initiator and a target.

3.1.73 physical reconnection: A physical reconnection is the result of a physical reconnect that exists from the assertion of the BSY signal in a SELECTION or RESELECTION phase. A physical reconnection ends with the BUS FREE phase (see 10.2) or a QAS REQUEST message (see 10.5.3).

3.1.74 planar: A cable media construction where the signal wires are substantially side by side. The cable media may contain twisted pairs or straight wires or any combination of twisted pairs/straight wires.

3.1.75 point-to-point: A subset of the multidrop architecture (see 4.3) where only two SCSI devices are attached within an allowed stub length of the terminators on a SCSI bus.

3.1.76 port: A single attachment to a SCSI bus from a SCSI device.

3.1.77 queue: The arrangement of tasks within a task set usually according to the temporal order that they were created.

3.1.78 queue tag: The parameter associated with a task that uniquely identifies it from other tagged tasks for a logical unit from the same initiator.

3.1.79 receiver: The circuitry used to detect the state of the bus.

3.1.80 request: The first step of a transaction invoking a confirmed service.

3.1.81 response: The third step of a four step confirmed service in reply to an indication.

3.1.82 SCSI address: The decimal representation of the unique address assigned to a SCSI device.

3.1.83 SCSI bus: All the conductors and connectors required to attain signal line continuity between every driver, receiver, and terminator for each signal.

3.1.84 SCSI device: A device containing at least one SCSI port and the means to connect its drivers and receivers to the bus.

3.1.85 SCSI ID: The bit-significant representation of the SCSI address.

3.1.86 signal assertion: The act of driving a signal to the true state.

3.1.87 signal negation: The act of performing a signal release or of driving a signal to the false state.

3.1.88 signal release: The act of allowing the cable terminators to bias the signal to the false state (by placing the driver in the high impedance condition).

3.1.89 single-ended (SE): A signalling alternative that uses SE (see 7.2) drivers and receivers (also see 3.1.21, differential).

3.1.90 single transition (ST): The latching of data only on the assertion edge of the REQ or ACK signals.

3.1.91 source (a signal): The act of either signal assertion, signal negation, or signal release.

3.1.92 SPI information unit: Data structures that encapsulate data, status, command, task attributes, iuCRC, and nexus information into various formats.

3.1.93 stub: Any electrical path connected to the bus that is not part of the bus path.

3.1.94 stubbed path: Path with stubs attached.

3.1.95 target: A SCSI device that receives SCSI commands and directs such commands to one or more logical units.

3.1.96 task: An object within the logical unit representing the work associated with a command or group of linked commands. A task consists of one initial connection and zero or more physical or logical reconnections, all pertaining to the task.

3.1.97 task manager: An agent within the device server that executes task management functions.

3.1.98 task management function: A task manager service that may be invoked by a task management message or by setting one of the task management flags in a SPI L_Q information unit to affect the execution of one or more tasks.

3.1.99 task set: A group of tasks within a device server, whose interaction is dependent on the task management, contingent allegiance and auto-contingent allegiance rules. See the SCSI Architecture Model-2 standard for a detailed definition of a task set.

3.1.100 transceiver: A device that implements both the SCSI bus receiver and driver functions.

3.1.101 transfer period: The negotiated time between edges of REQ or ACK that latch data. For ST the transfer period is measured from assertion edge of the REQ or ACK signal to the next assertion edge of the signal. For DT the transfer period is measured from a transition edge of the REQ or ACK signal to the next transition edge of the signal.

3.1.102 transfer rate: The negotiated megatransfers per second.

3.1.103 upper level protocol: Any protocol executed through services provided by a lower level protocol.

3.1.104 vendor-specific: Something (e.g., a bit, field, code value) that is not defined by this standard and may be used differently in various implementations.

3.1.105 zero: A false signal value or a false condition of a variable.

3.2 Symbols and abbreviations

≠ or NE	not equal
≤ or LE	less than or equal to
±	plus or minus
≈	approximately
x	multiply
+	add
-	subtract
< or LT	less than
= or EQ	equal
> or GT	greater than
≥ or GE	greater than or equal to
ACA	auto-contingent allegiance (see 3.1.7)
AWG	American wire gauge
CA	Contingent allegiance (see 3.1.13)
CMOS	Complementary metal oxide semiconductor
CRC	Cyclic Redundancy Check (see 3.1.14)
DT	Double transition (see 3.1.22)
DUT	Device under test
EMI	Electromagnetic interference
EMC	Electromagnetic compatibility
ESD	Electrostatic discharge
HVD	High voltage differential
IDC	Insulation displacement contact
ISI	Intersymbol interference
iuCRC	Information unit CRC
LSB	Least significant bit
LUN	Logical unit number
LVD	Low voltage differential
MSB	Most significant bit
MSE	Multimode single ended (see 3.1.54)
pCRC	Parallel CRC
PPR	Parallel protocol request
NEXT	Near end crosstalk
SCSI	Small Computer System Interface
QAS	Quick Arbitration and Selection
SCSI-2	Small Computer System Interface - 2
SCSI-3	Small Computer System Interface - 3
SDTR	Synchronous data transfer request

SE	Single-ended (see 3.1.89)
ST	Single transition (see 3.1.90)
WDTR	Wide data transfer request

3.3 Keywords

3.3.1 expected: A keyword used to describe the behavior of the hardware or software in the design models assumed by this standard. Other hardware and software design models may also be implemented.

3.3.2 invalid: A keyword used to describe an illegal or unsupported bit, byte, word, field or code value. Receipt of an invalid bit, byte, word, field or code value shall be reported as an error.

3.3.3 mandatory: A keyword indicating an item that is required to be implemented as defined in this standard to claim compliance with this standard.

3.3.4 may: A keyword that indicates flexibility of choice with no implied preference.

3.3.5 may not: Keywords that indicates flexibility of choice with no implied preference.

3.3.6 obsolete : A keyword indicating that an item was defined in prior SCSI standards but has been removed from this standard.

3.3.7 optional: A keyword that describes features that are not required to be implemented by this standard. However, if any optional feature defined by this standards is implemented, it shall be implemented as defined in this standard.

3.3.8 reserved: A keyword referring to bits, bytes, words, fields and code values that are set aside for future standardization. Their use and interpretation may be specified by future extensions to this or other standards. A reserved bit, byte, word or field shall be set to zero, or in accordance with a future extension to this standard. Recipients are not required to check reserved bits, bytes, words or fields for zero values. Receipt of reserved code values in defined fields shall be reported as an error.

3.3.9 shall: A keyword indicating a mandatory requirement. Designers are required to implement all such requirements to ensure interoperability with other products that conform to this standard.

3.3.10 should: A keyword indicating flexibility of choice with a preferred alternative; equivalent to the phrase "it is recommended".

3.4 Conventions

Certain words and terms used in this American National Standard have a specific meaning beyond the normal English meaning. These words and terms are defined either in clause 3 or in the text where they first appear. Names of signals, phases, messages, commands, statuses, sense keys, additional sense codes, and additional sense code qualifiers are in all uppercase (e.g., REQUEST SENSE), names of fields are in small uppercase (e.g., STATE OF SPARE), lower case is used for words having the normal English meaning.

Fields containing only one bit are usually referred to as the name bit instead of the name field.

Numbers that are not immediately followed by lower-case b or h are decimal values.

Numbers immediately followed by lower-case b (xxb) are binary values.

Numbers immediately followed by lower-case h (xxh) are hexadecimal values.

Decimals are indicated with a comma (e.g., two and one half is represented as 2,5).

Decimal numbers having a value exceeding 999 are represented with a space (e.g., 24 255).

An alphanumeric list (e.g., a,b,c or A,B,C) of items indicate the items in the list are unordered.

A numeric list (e.g., 1,2,3) of items indicate the items in the list are ordered (i.e., item 1 must occur or complete before item 2).

In the event of conflicting information the precedence for requirements defined in this standard is:

1) text,
2) tables, then
3) figures.

3.5 Notation for Procedures and Functions

Procedure Name ([input:1a|input:1b|input:1c][,input:2a+input:2b]...[input:n]||
[output:1][,output:2]...[output:n])

Where:

Procedure Name:	A descriptive name for the function to be performed.
"(...)":	Parentheses enclosing the lists of input and output arguments.
input:1a|input:1b|...	A number of arguments of which only one shall be used in any single procedure
input:1, input:2, ...:	A comma-separated list of names identifying caller-supplied input data objects.
output:1, output:2, ...:	A comma-separated list of names identifying output data objects to be returned by the procedure.
"||":	A separator providing the demarcation between inputs and outputs. Inputs are listed to the left of the separator; outputs, if any, are listed to the right.
"[...]":	Brackets enclosing optional or conditional parameters and arguments.
"|":	A separator providing the demarcation between a number of arguments of which only one shall be used in any single procedure.
"+":	A collection of objects presented to a single object. No ordering is implied.

4 General

4.1 General overview

This standard defines the cables, connectors, signals, transceivers, and protocol used to interconnect parallel SCSI devices and the services provided to the application client.

4.2 Cables, Connectors, Signals, Transceivers

SCSI parallel interface devices may be implemented with either 50, 68, or 80 pin connectors.

Table 1 defines the bus modes and transfer rates supported with the various transceivers defined within this standard.

Table 1 - Transceiver/speed support map

Transceiver Type	Data Latching (ST/DT)	Maximum transfer rate						
		Async.	Fast-5	Fast-10	Fast-20	Fast-40	Fast-80	Fast-160
SE	ST	yes	yes	yes	yes	no	no	no
MSE (Note)	ST	yes	yes	yes	yes	no	no	no
LVD	ST	yes	yes	yes	yes	yes	no	no
	DT	no	no	yes	yes	yes	yes	yes
Key: yes = Transceiver/speed combination supported by this standard. No =Transceiver/speed combination not supported by this standard.								
Note-MSE is identical to SE except for the requirements in 7.4, table 15, and table 16.								

SCSI devices may connect to the bus via 8-bit or 16-bit ports. The 8-bit ports shall connect to a bus with an A cable or equivalent (see 5). The 16-bit ports shall connect to a bus with a P cable or equivalent (see 5).

4.3 Physical architecture of bus

The position of the drivers, receivers, and terminators for a SE bus are shown in figure 2 and for a differential bus are shown in figure 3. The electrical properties of the drivers and receivers are all measured at the stub connections (see figure 4). Unless otherwise noted, all voltages are with respect to the signal ground of the SCSI device.

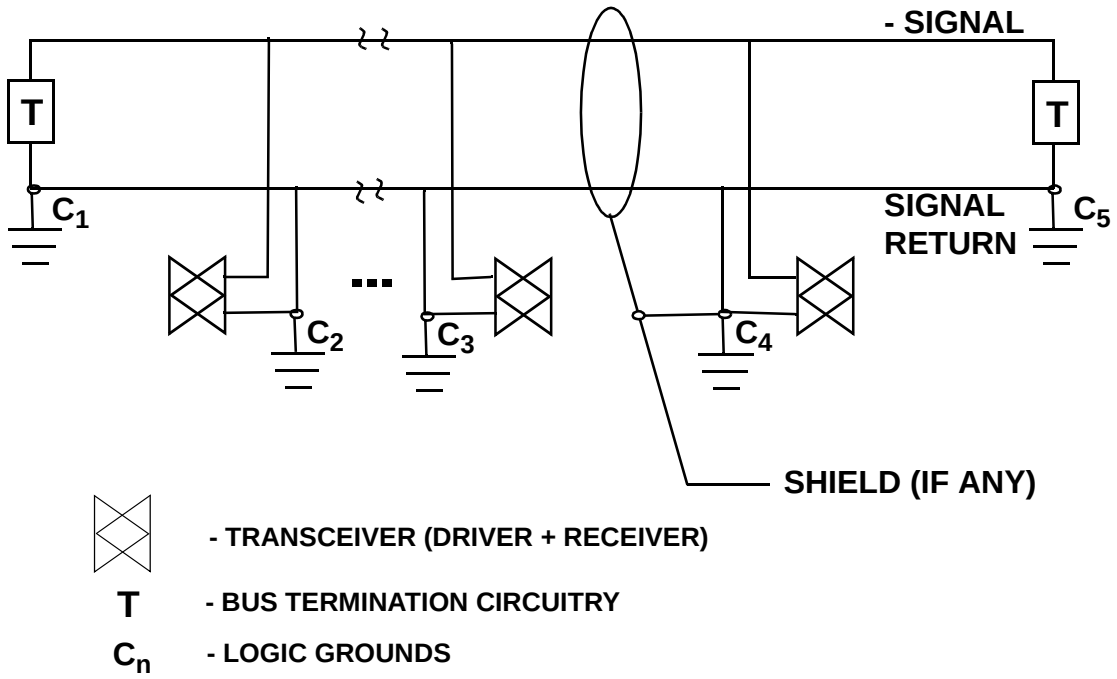


Figure 2 - SE SCSI bus

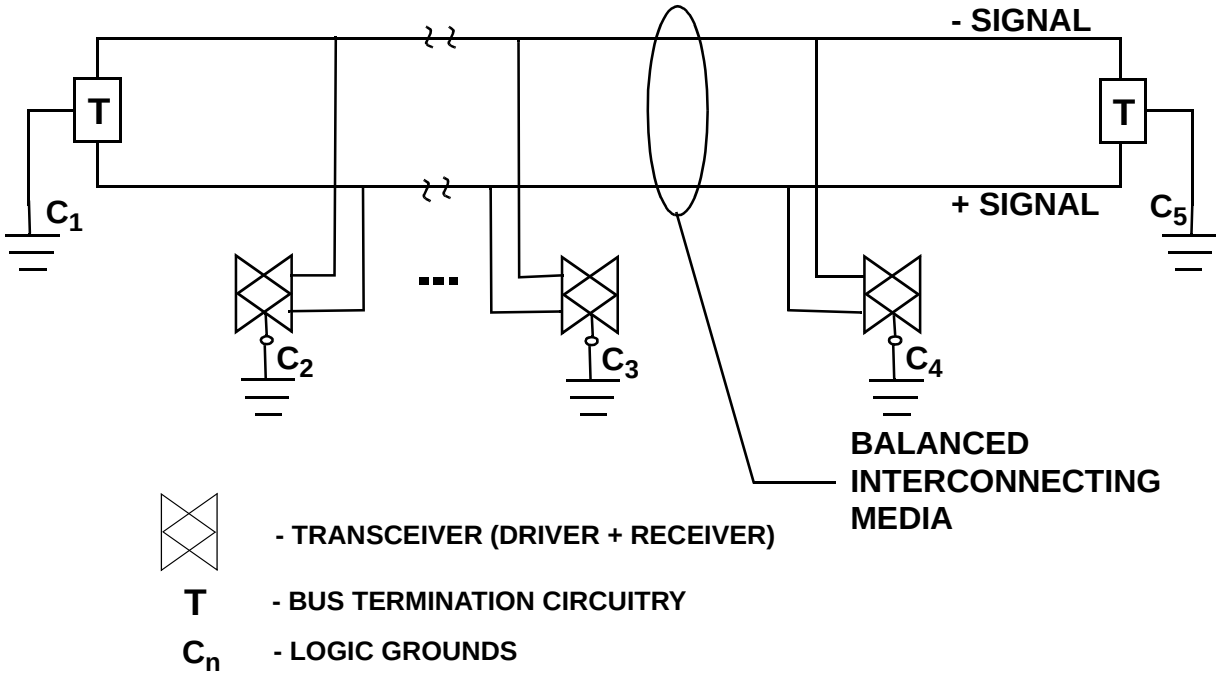


Figure 3 - Differential SCSI bus

4.4 Physical topology details and definitions

The SCSI bus is a multidrop architecture described in 4.3. Other details important to ensure the proper

operation of this topology are described in this subclause.

The SCSI bus consists of all the conductors and connectors required to attain signal line continuity between every driver, receiver, and terminator for each signal. The electrical connection directly between the two terminators forms the bus path. Any electrical path that is not part of the bus-path is a stub. The point where a stub meets the bus path is termed the stub connection.

Figure 4 shows examples of connectors, bus paths, stubs, and stub connections.

SCSI bus connectors are any connector, defined within this standard, used to create the SCSI bus. SCSI bus connectors are defined by their function and by their physical placement.

The functional definitions are:

a) connectors used to provide part of the bus-path are labeled bus-path connectors, and
b) connectors used to provide part of a stub are labeled stub connectors.

Common physical placement definitions are:

a) connectors physically part of SCSI devices are labeled device connectors,
b) connectors physically part of cables, backplanes, or other non-device conductors are labeled cable connectors,
c) connectors physically part of terminators are labeled terminator connectors,
d) connectors that provide entry and exit points to and from enclosures are labeled enclosure connectors, and
e) other physical placements may be used.

SCSI bus connectors (e.g., device stub connector, terminator bus path connector) referred to in this standard use both the functional definition and a physical placement.

The portion of the stub contained within the stub connector that has the stub connection may be ignored.

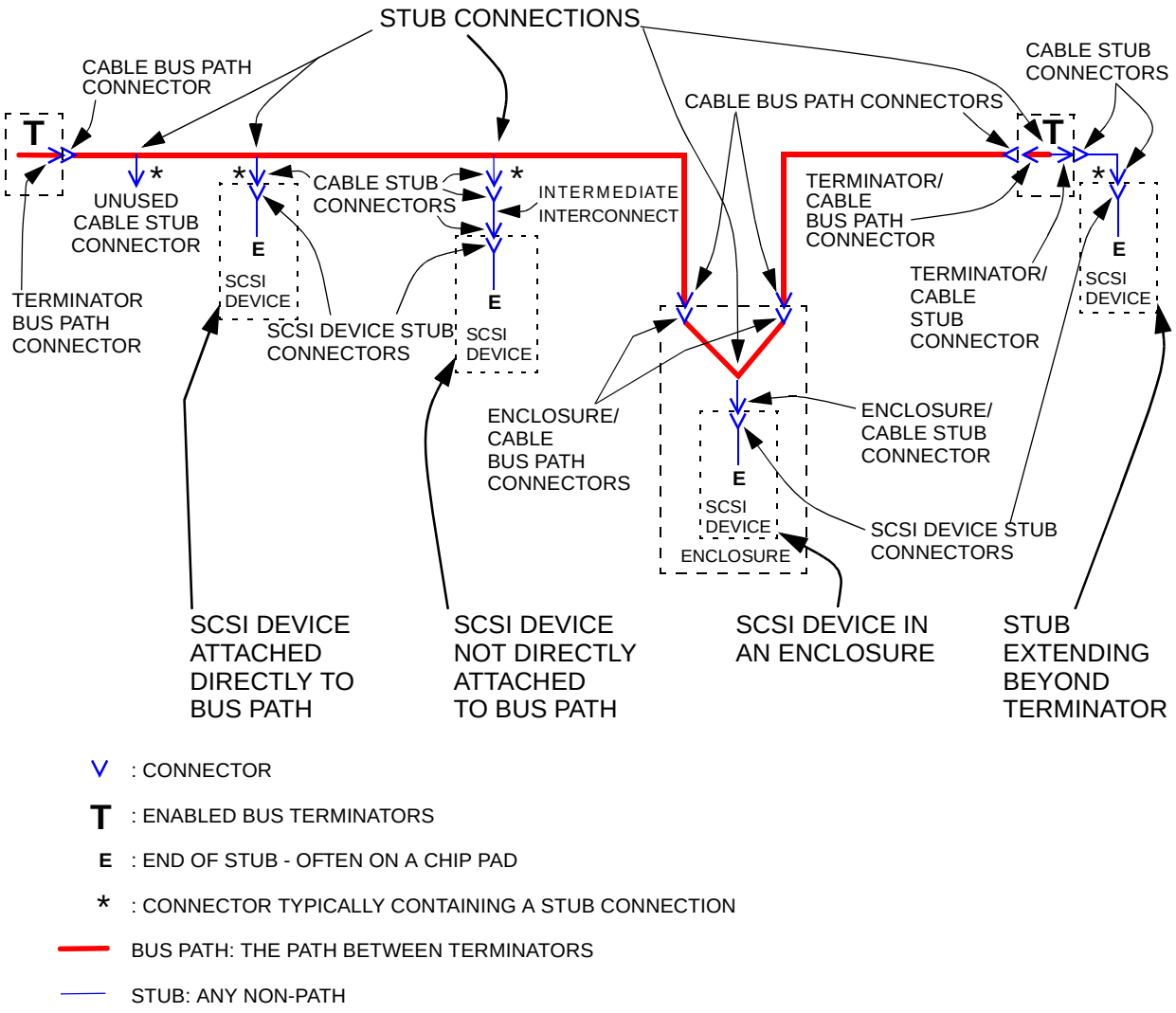


Figure 4 - SCSI bus topology details

If an intermediate interconnection is added to connect the SCSI device to the bus path this additional interconnect (including its connectors) and the SCSI device contribute to the stub and bus loading. In system implementations that use an intermediate interconnect the parameters specified in this standard at the SCSI device connector shall apply at the stub connection.

NOTE 3 - Any extensions of the connection beyond the terminator as shown in the right side of figure 4 should be minimized or avoided as that extension produces stubs and bus loading.

NOTE 4 - In order to support daisy-chain connections, SCSI devices that use shielded connectors should provide two shielded SCSI device connectors on the SCSI device enclosure. Inside the enclosure the cable should be looped from one shielded connector to the other. The loop should pass the connecting point to the transceivers within the enclosure in such a manner that stub lengths are minimized. The length of the cable within the SCSI device enclosure is included when calculating the total cable length of the SCSI bus. (see figure 4)

4.5 Bus loading

Bus loading is the electrical current flowing through the stub connection for lines that are not being driven

by the attached SCSI device. The bus termination circuitry also provides bus loading. Bus loading shall appear capacitive to A.C. signals and may also have a D.C. leakage component. The stub capacitance is caused by electrical paths and components within the stub. The leakage is caused by imperfect insulation of plus and minus signals and by components attached to the paths within the stub. The capacitive current loading is specified by the value of the capacitances at the plus and minus signals rather than by the value of the current.

Bus termination circuitry bus loading is the capacitance measured at the terminator bus-path connector. Any D.C. leakage within enabled terminators is part of the performance requirements in 7.2.1, 7.3.1, and 7.4.1 and does not constitute bus loading.

Bus termination loading is separate from bus loading. SCSI devices containing enabled bus termination shall present maximum loading at the stub connection that is the sum of the maximum allowed termination loading and the maximum allowed bus loading. See 4.6 for requirements of disabled termination circuitry.

For stub connections within an allowed stub length from enabled bus termination circuitry, the maximum bus loading allowed is the sum of the maximum bus termination loading and the maximum bus loading. If the enabled terminators are within a SCSI device and if either the bus termination loading or the bus loading is less than the maximum allowed, the other entity may increase its loading as long as the total for both entities does not exceed the maximum allowed.

4.6 Termination requirements

The SCSI bus termination defines the ends of the SCSI bus. Bus termination is required to set the negated state when no SCSI device is driving (also called biasing) and to match the impedance to that of the interconnect media. A termination circuit is providing bus termination only when it is delivering the performance requirements for biasing and impedance matching. Such a termination circuit is said to be enabled when it is providing the bus termination.

Terminator circuits may also be in a disabled state, when they are not providing any of the termination functions of bias and impedance matching. One way of disabling a terminator is to disconnect all the signal lines (optionally including DIFFSENS) by an electronic switch. Such a terminator circuit is called a switchable terminator.

Disabled terminators count as SCSI devices in terms of bus loading if they are individually attached to the bus. If they are contained within a SCSI device the disabled terminators become part of the SCSI device load budget for that SCSI device.

4.7 SCSI device Addressing

The number of SCSI devices that may be addressed depends on the width of the data path of the bus; an 8-bit data path allows up to 8 SCSI devices to be addressed, and a 16-bit data path allows up to 16 SCSI devices to be addressed. However, the number of SCSI devices that may be connected to the bus is dependent on several factors (e.g., bus length, data transfer rates, capacitance loading of the SCSI device) that are described throughout this standard.

4.8 Data transfers

This standard defines optional methods of latching data from the REQ and ACK signals depending on whether ST DATA phases or DT DATA phases are being used for information transfers as shown in figure 5. Data shall only be latched on the asserting edge of the REQ or ACK signal except in DT DATA phases. When DT DATA phases are used, data shall be latched on both the asserting edge and the negating edge of the REQ or ACK signal.

Regardless of whether ST or DT transfers are enabled the negotiated transfer period sets the maximum

rate that data is clocked at in megatransfers per second. As a result, the time from rising edge to rising edge for REQ and ACK signals for the same transfer rate is twice as long for a DT transfer as it is for an ST transfer. An example of a negotiated transfer period of 25 ns with ST transfers is shown in figure 6. While the same 25 ns negotiated transfer period with DT transfers is shown in figure 7. The only timing difference between ST and DT is that the rising edge to rising edge time for DT is 50 ns while ST is 25 ns. In both cases data is transferred at 25 ns intervals.

Figure 8 shows an example of what paced transfers look like at the receiving SCSI devices connector with a negotiated transfer period of 6.25 ns. There is no difference as to when data is latched on paced transfers, however, the relationship between the data and REQ or ACK is required to be adjusted in the SCSI devices receiver to match the synchronous transfers DT Data shown in figure 5.

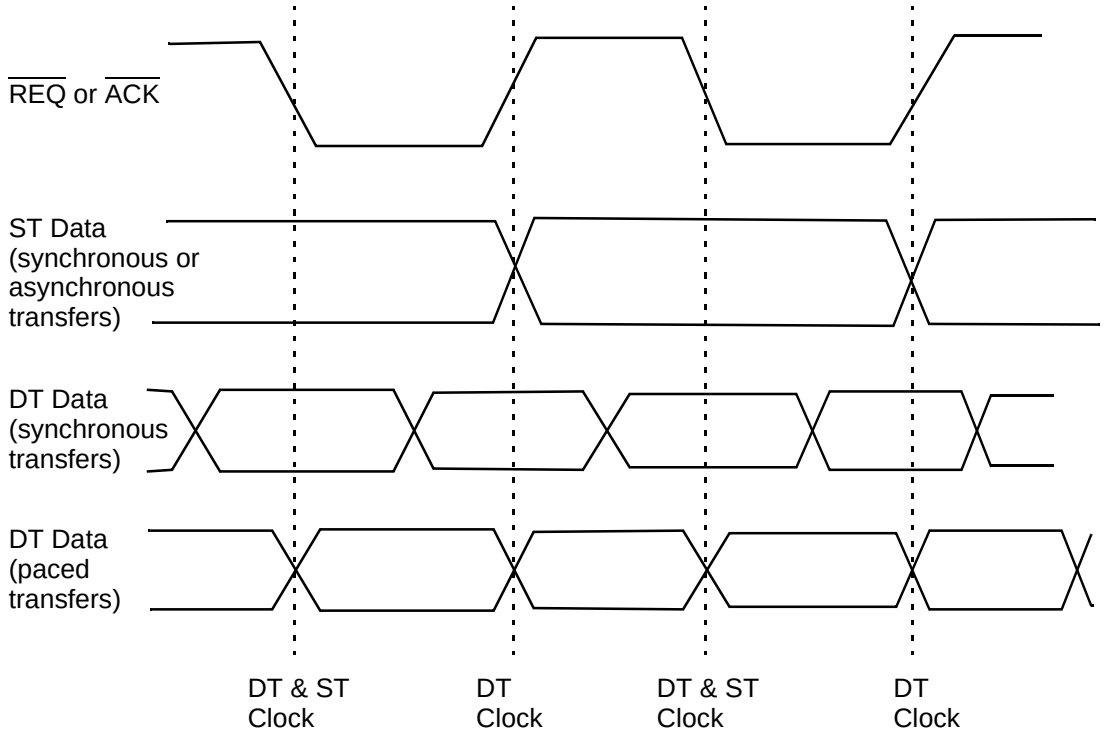
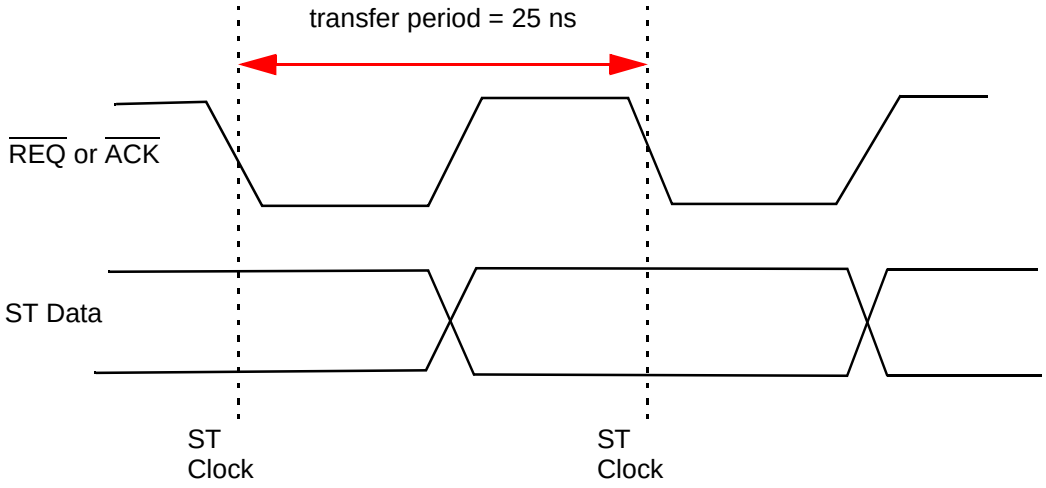
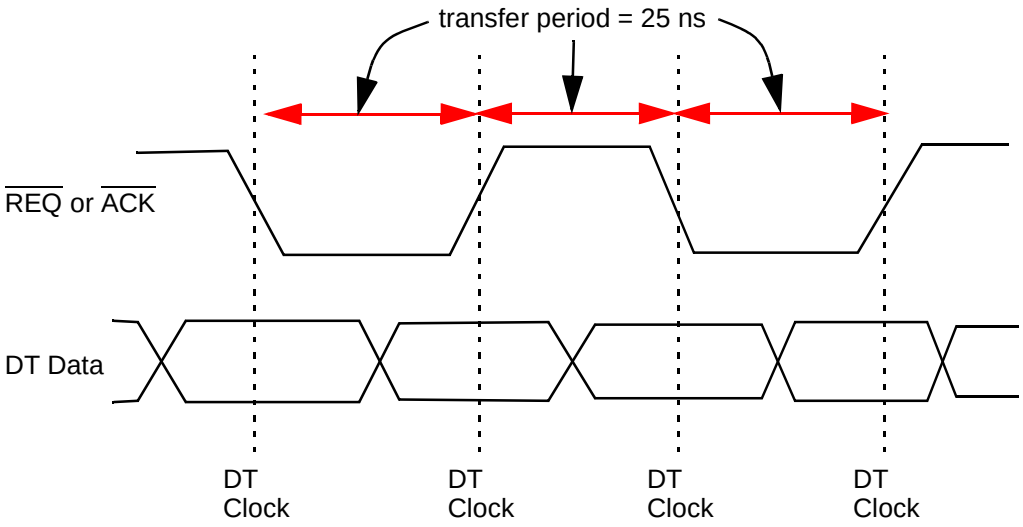


Figure 5 - ST latching data vs. DT latching data



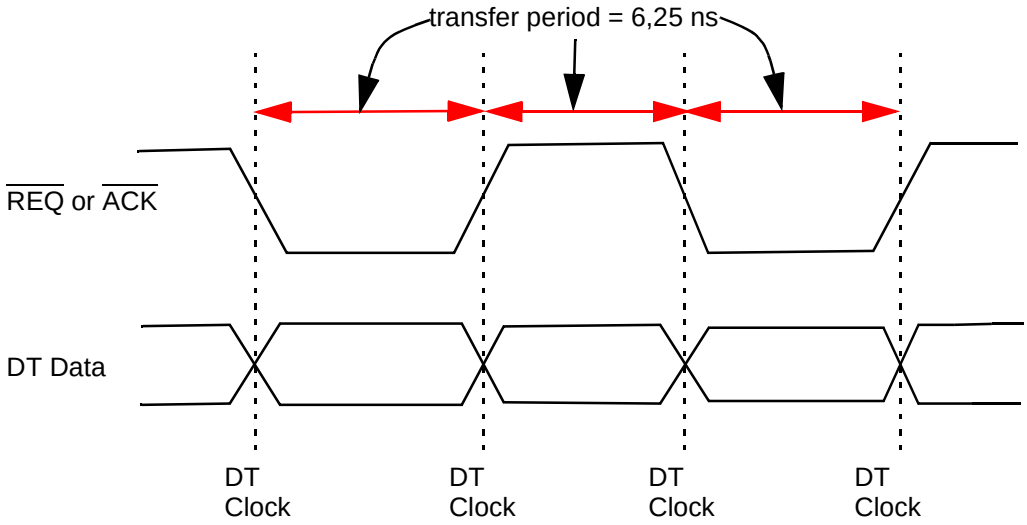
Example: A negotiated transfer period of 25 ns equates to a transfer rate of 40 megatransfers per second.

Figure 6 - Synchronous ST data transfer example



Example: A negotiated transfer period of 25 ns equates to a transfer rate of 40 megatransfers per second.

Figure 7 - Synchronous ~~ST~~ DT data transfer example



Example: A negotiated transfer period of 6,25 ns equates to a transfer rate of 160 megatransfers per second.

Figure 8 - ~~Synchronous~~ Paced DT data transfer example

Editors Note 1 - GOP: Think about putting a diagram that contains a connector - deskew - optional AAF - data detector.

4.9 Data transfers

4.9.1 Data transfer modes

4.9.1.1 Asynchronous transfers

SCSI parallel interface devices default to 8-bit asynchronous transfers.

The 8-bit asynchronous transfers are used for all information transfers except DATA phases. ST DATA phases may use 8-bit or 16-bit asynchronous transfers when a wide transfer agreement (see 10.8.5) is in effect. Asynchronous transfers are not permitted when DT DATA phases are enabled.

4.9.1.2 Synchronous transfers

Synchronous transfers shall only be used in DATA phases when a synchronous transfer agreement is in effect (see 10.8.5).

ST DATA phases shall transfer data using synchronous transfers when a ST DATA phase enabled agreement is in effect. ST DATA phases may use 8-bit or 16-bit synchronous transfers when a wide transfer agreement (see 10.8.5) is in effect.

DT DATA phases shall transfer data using synchronous transfers when a DT DATA phase enabled agreement is in effect (see 10.8.5). DT DATA phases shall only use 16-bit synchronous transfers.

4.9.1.3 Paced transfers

Paced transfers shall only be used in DT DATA phases when a ~~paced transfer~~ fast-160 agreement is in effect (see 10.8.5). DT DATA phases shall only use 16-bit paced transfers.

4.9.2 ST DATA phase parallel transfers

The format of data transmitted during ST DATA phases consists of data and protection. Parity is used to protect the data (see 11.2).

4.9.3 DT DATA phase parallel transfers

4.9.3.1 DT DATA phase parallel transfers format

The format of the data transmitted during DT DATA phases is dependent on the negotiated protocol option. If data group transfers are enabled then all data and protection are transmitted in data groups. If information unit transfers are enabled then all nexus, task management, task attribute, command, data, and protection are transmitted in SPI information units.

4.9.3.2 Data group transfers

Data group transfers are permitted when a synchronous transfer agreement is in effect. Data group transfers are not permitted when an asynchronous transfer agreement or a paced transfer agreement is in effect (see 10.8.5).

When using data group transfers each DT DATA IN phase and DT DATA OUT phase contains one or more data groups. A data group consists of a non-zero length data field containing an even number of bytes (see 16.3.4), followed by a pad field when pad bytes are needed, and then followed by a pCRC field. The number of bytes transferred within a data group shall always be a multiple of four.

If the number of bytes in the data field is not a multiple of four the transmitting SCSI device shall place two pad bytes into the pad field. If the number of bytes in the data field is a multiple of four the transmitting SCSI device shall omit the pad field. Regardless of the number of bytes in the data field the pCRC field shall be the last four bytes of the data group.

The value of the pad bytes within the pad field is vendor specific.

During DT DATA IN phase if the number of bytes in a data field is not a multiple of two bytes, then after sending the pad and pCRC fields the target shall change to MESSAGE IN phase and send an IGNORE WIDE RESIDUE message (16.3.4) with the NUMBER OF BYTES TO IGNORE field set to 01h.

During DT DATA OUT phase if a target requests a pCRC field prior to the last data field of a task, the initiator shall transmit an even number of bytes in that data field.

The pCRC shall be used to protect all data group transfers. The SCSI device transmitting data sends the necessary pad field(s) and a pCRC field at a point determined by the target.

4.9.3.3 Information unit transfers

Information unit transfers are permitted when a synchronous transfer agreement or a paced transfer agreement is in effect. Information unit transfers are not permitted when an asynchronous transfer agreement is in effect (see 10.8.5).

During information unit transfers each DT DATA IN phase and DT DATA OUT phase contains one or more SPI information units. The number of bytes transferred within a SPI information unit shall always be a multiple of four.

If the number of bytes in the SPI information unit is not a multiple of four the transmitting SCSI device shall transmit one, two, or three pad bytes before transmitting an iuCRC. If the number of bytes in the SPI information unit is a multiple of four the transmitting SCSI device shall not transmit any pad bytes. Regardless of the number of bytes in the SPI information unit the last four bytes of the SPI information unit shall be an iuCRC.

The value of the pad bytes is vendor specific.

The iuCRC shall be used to protect all SPI information units. The SCSI device that originates the SPI information unit sends the necessary pad byte(s) and iuCRC field(s).

An iuCRC interval may also be specified. The iuCRC interval specifies the number of bytes transferred before pad bytes (if any) and the iuCRC is transferred within SPI data information units and SPI data stream information units. A SPI data information unit or a SPI data stream information unit may contain zero or more iuCRC intervals depending on the values specified in the SPI L_Q information unit. At a minimum there shall be at least one iuCRC at the end of each SPI data information unit and SPI data stream information unit regardless of the size of the iuCRC interval.

The iuCRC interval is required to be a multiple of two, however, if it is not a multiple of four then two pad bytes shall be transmitted before each iuCRC interval, iuCRC is transmitted.

SPI data stream information units may be used to transfer data to or from a SCSI device. Support of data streaming is mandatory during DT DATA OUT phases and is optional during DT DATA IN phases. The use of data streaming during DT DATA IN phases is part of the negotiated agreement between two SCSI devices and is referred to as read streaming within this standard.

A target, while steaming data, may give an indication that the stream of SPI data stream information units are about to end while still sending the current SPI data stream information unit. This early warning is called flow control. The use of flow control in DT DATA OUT phases is part of the negotiated agreement between two SCSI devices and is referred to as write flow control within this standard. The use of flow control in DT DATA IN phases is combined with the negotiation agreement on data streaming during DT DATA IN phases.

NOTE 5 - Because the DT DATA IN phase flow control and read streaming require the same SCSI bus signals to be controlled in the same way at the same time this standard makes no further reference to DT DATA IN phase flow control, however, that functionality is implied by the term read streaming.

4.10 Negation

Editors Note 2 - GOP: Put all the rules for target/initiator negotiations in this section including the REQ(ACK) offset definition from 99-295r5.

~~**REQ(ACK) offset:** for Fast =<80 transfers the number of REQ assertions that may be sent by the target in advance of the number of ACK assertions received from the initiator, establishing a pacing mechanism. For Fast 160 during Data Out transfers the number of REQ transitions that may be sent by the target in advance of the number of P1 enabled data transfers received from the initiator, establishing a data pacing mechanism. For Fast 160 during Data In transfers the number of P1 enabled data transfers that may be sent by the target in advance of ACK transitions received from the initiator, establishing a data pacing mechanism.~~

REQ/ACK offset: For ST DATA transfers the REQ/ACK offset is the number of REQ assertions that may be sent by the target in advance of the number of ACK assertions received from the initiator.

For DT DATA transfers not using paced transfers the REQ/ACK offset is the number of REQ transitions that may be sent by the target in advance of the number of ACK transitions received from the initiator.

For paced DT DATA IN transfers the REQ/ACK offset is the number of data valid state REQ assertions (see 10.8.4.3) that may be sent by the target in advance of ACK assertions received from the initiator.

For paced DT DATA OUT transfers the REQ/ACK offset is the number of REQ assertions that may be sent

by the target in advance of the number of data valid state ACK assertions (see 10.8.4.3) received from the initiator.

4.11 Protocol

This standard describes a SCSI device's behavior in terms of functional levels, service interfaces between levels and peer-to-peer protocols. For a full description of the model used in this standard see the SCSI Architecture Model-2 standard. Figure 9 shows the model as it appears from the point of view of this standard.

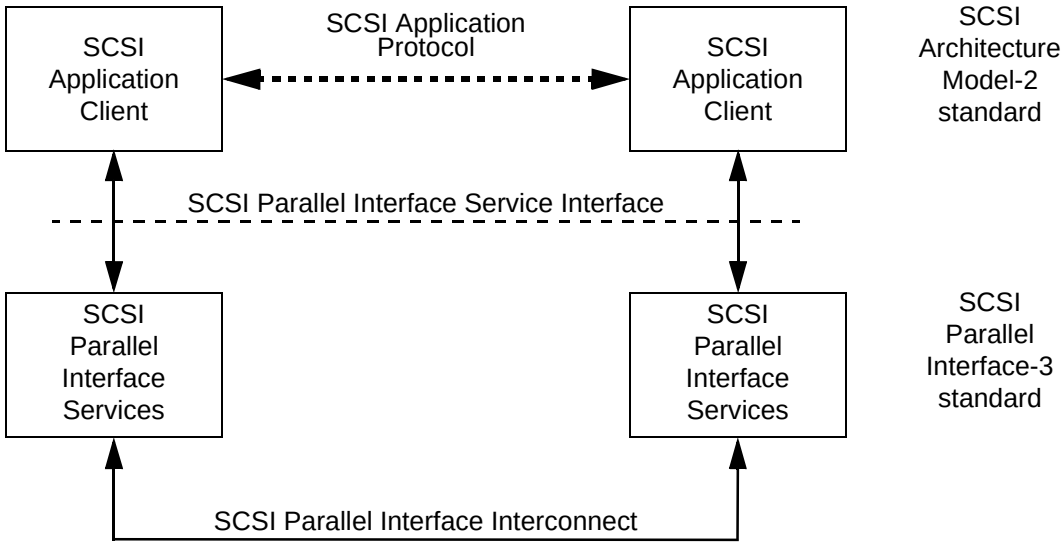


Figure 9 - SCSI Parallel Interface service reference mode

Services between service levels are either four step confirmed services or two step confirmed services. A four step confirmed service consists of a service request, indication, response, and confirmation. A two step confirmed service consists of a service request and confirmation.

Figure 10 shows the service and protocol interactions for a four step confirmed service.

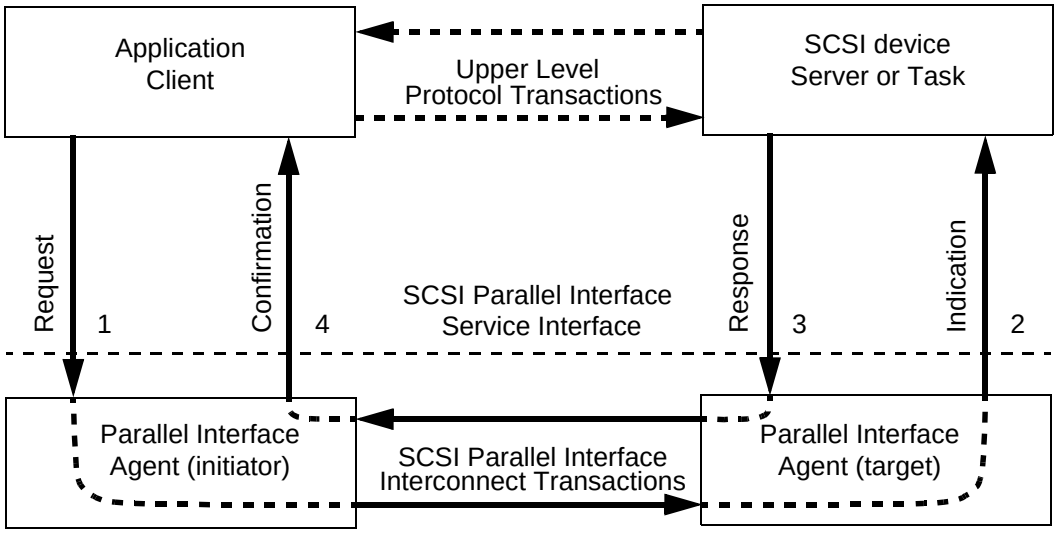


Figure 10 - Model for a four step confirmed service

The SCSI parallel interface four step confirmed service protocol consists of the following interactions:

a) A request to the initiator parallel interface agent to invoke a service;
b) An indication from the target parallel interface agent notifying the SCSI device server or task manager of an event;
c) A response from the SCSI device server or task manager in reply to an indication;
d) A confirmation from the initiator parallel agent upon service completion.

Only application clients shall request a four step confirmed service be invoked.

Figure 11 shows the service and protocol interactions for a two step confirmed service.

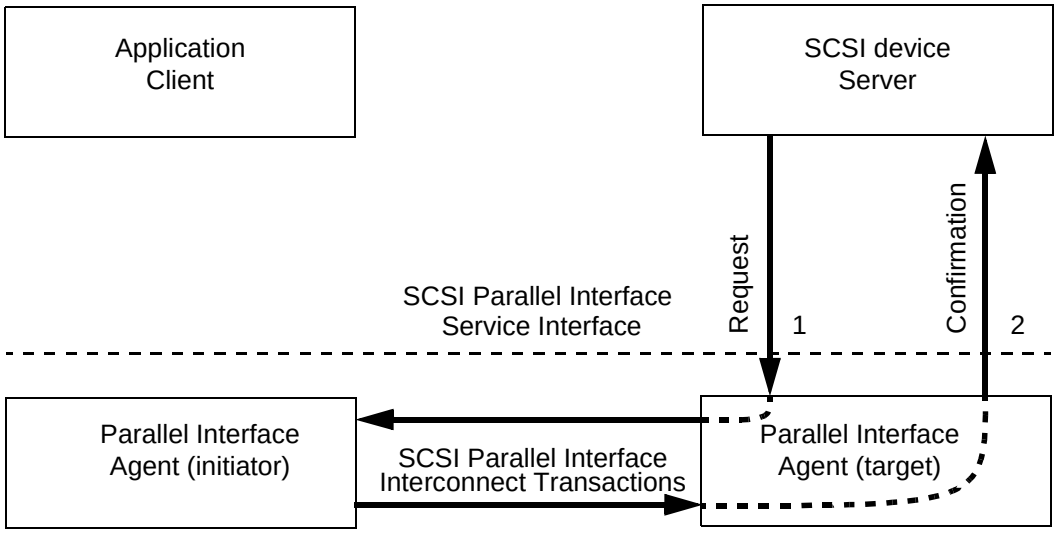


Figure 11 - Model for a two step confirmed service

The SCSI parallel interface two step confirmed service interface consists of the following interactions:

a) A request to the target parallel interface agent to invoke a service;
b) A confirmation from the target parallel interface agent upon service completion.

Only SCSI device servers shall request a two step confirmed service be invoked.

5 SCSI parallel interface connectors

5.1 SCSI parallel interface connectors overview

Two types of connectors are defined: nonshielded and shielded. The nonshielded connectors are typically used within an enclosure. The shielded connectors are typically used for external applications where electromagnetic compatibility (EMC) and electrostatic discharge (ESD) protection may be required. Either type of connector may be used with the single-ended or differential transceivers.

The connector mechanical drawings conform to the ANSI Y14.5M-1994 (geometric form tolerancing) standard.

This standard defines all the supported SCSI device connectors. The 80-contact alternative 4 non-shielded SCSI device connector and the 68-contact alternative 4 shielded SCSI device connector are defined by reference to IEC standards (see 2).

The alternative 1 nonshielded, alternative 3 nonshielded, alternative 1 shielded, and alternative 3 shielded connectors shall have contact geometry and normal force sufficient to pass the following test:

a) Measure contact resistances of the connectors being evaluated using a test procedure for low-level contact resistance. Use IEC 512-2 (low-level contact resistance test procedure for electronic connectors) as a reference procedure. Record measurements as initial contact resistances;
b) Mate and unmate connectors 50 cycles;
c) Measure contact resistance in accordance with item (a) above (this is an optional step);
d) Expose mated connectors to mixed flowing gas consisting of 10 parts per billion (ppb) of chlorine, 10 ppb of hydrogen sulfide, 200 ppb of sulfur dioxide, and 200 ppb of nitrogen dioxide for 20 days at 70 % relative humidity and 30 ^{o}C. Use IEC 512-11-7 (standard practice for conducting mixed flowing gas environmental tests) as a reference procedure;
e) Remove connectors from the mixed flowing gas, remeasure contact resistance in accordance with item a) above. Any contact with an increase of 15 milliohms or greater is a failure.

The resistance shall be measured using a four-point dry-circuit method directly across the mated contact.

5.2 Nonshielded connector

5.2.1 Nonshielded connector alternative 1 - A cable

The alternative 1 nonshielded SCSI device connector for the A cable shall be a 50-conductor connector consisting of two rows of 25 female contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 12. The nonmating portion of the connector is shown for reference only.

The alternative 1 nonshielded mating connector for the A cable shall be a 50-conductor connector consisting of two rows of 25 male contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 13. The nonmating portion of the connector is shown for reference only.

5.2.2 Nonshielded connector alternative 2 - A cable

The alternative 2 nonshielded SCSI device connector for the A cable shall be a 50-conductor connector consisting of two rows of 25 male contacts with adjacent contacts 2,54 mm (0,1 in) apart as shown in figure 14. A shroud and header body should be used. The non-mating portion of the connector is shown for reference only.

The alternative 2 nonshielded mating connector for the A cable shall be a 50-conductor connector consisting of two rows of 25 female contacts with adjacent contacts 2,54 mm (0,1 in) apart as shown in figure 15. It is recommended that keyed connectors be used.

5.2.3 Nonshielded connector alternative 3 - P cable

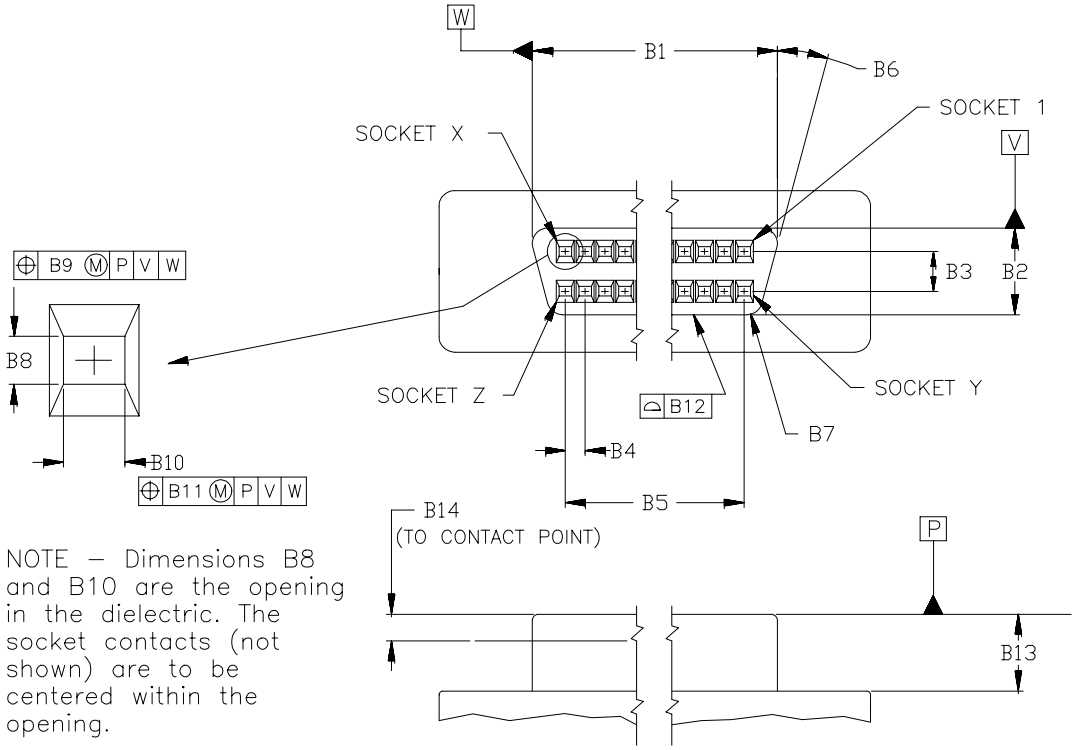
The alternative 3 nonshielded SCSI device connector for the P cable shall be a 68-conductor connector consisting of two rows of 34 female contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 12. The nonmating portion of the connector is shown for reference only.

The alternative 3 nonshielded mating connector for the P cable shall be a 68-conductor connector consisting of two rows of 34 male contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 13. The nonmating portion of the connector is shown for reference only.

5.2.4 Nonshielded connector alternative 4

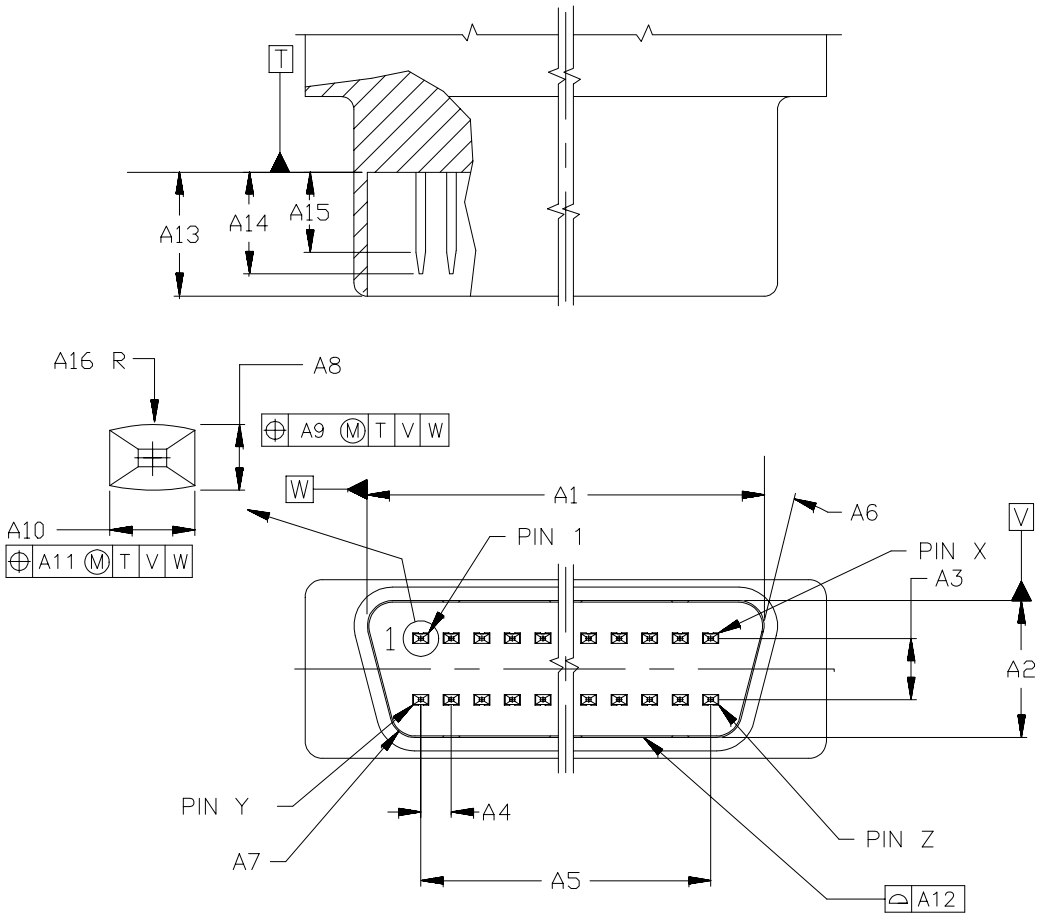
The alternative 4 nonshielded SCSI device connector for the P-cable shall be an 80-conductor connector consisting of two rows of ribbon contacts spaced 1,27 mm (0,05 in) apart, as shown in figure 16 and figure 17. For the detailed dimensional drawings of this connector see the SCA-2 EIA standard EIA-700A0AE.

The alternative 4 nonshielded mating connector for the P-cable shall be an 80-conductor connector consisting of two rows of ribbon contacts spaced 1,27 mm (0,05 in) apart, as shown in figure 16 and figure 17. For the detailed dimensional drawings of this connector see the SCA-2 EIA standard EIA-700A0AE and SCA-2 Unshielded Connections SFF-8451.



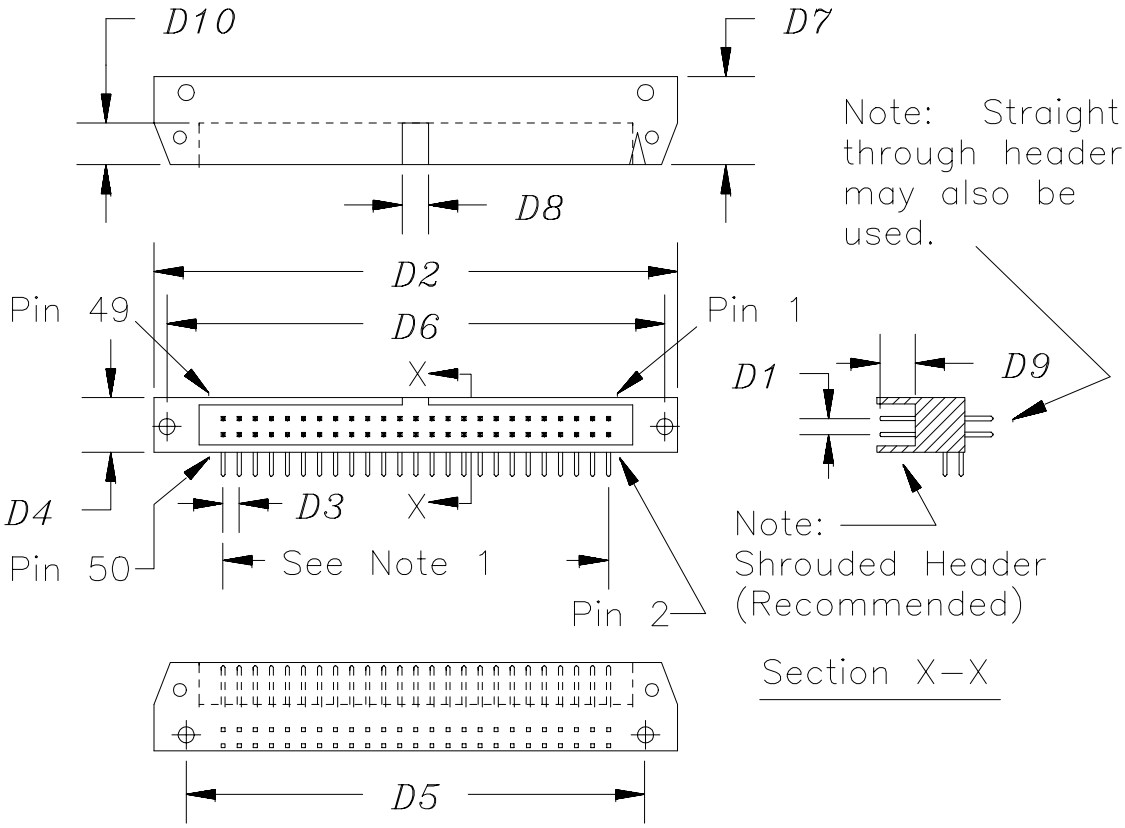
Dimensions	50 Position		68 Position	
	Millimeters	Inches	Millimeters	Inches
B1	34,70	1,366	46,13	1,816
B2	5,54	0,218	5,54	0,218
B3	2,54	0,100	2,54	0,100
B4	1,27	0,050	1,27	0,050
B5	30,48	1,200	41,91	1,650
B6	15°	15°	15°	15°
B7	1,00 R	0,039 R	1,00 R	0,039 R
B8	0,61± 0,05	0,024± 0,002	0,61± 0,05	0,024± 0,002
B9	0,15	0,006	0,15	0,006
B10	0,86± 0,10	0,034± 0,004	0,86± 0,10	0,034± 0,004
B11	0,15	0,006	0,15	0,006
B12	0,05	0,002	0,05	0,002
B13	5,00± 0,13	0,197± 0,005	5,00± 0,13	0,197± 0,005
B14	1,75 MAX	0,069 MAX	1,75 MAX	0,069 MAX
SOCKET X	25		34	
SOCKET Y	26		35	
SOCKET Z	50		68	

Figure 12 - 50/68-contact alternative 1/alternative 3 nonshielded SCSI device connector
(A cable/P cable)



Dimensions	50 Position		68 Position	
	Millimeters	Inches	Millimeters	Inches
A1	34,85	1,372	46,28	1,822
A2	5,69	0,224	5,69	0,224
A3	2,54	0,100	2,54	0,100
A4	1,27	0,050	1,27	0,050
A5	30,48	1,200	41,91	1,650
A6	15°	15°	15°	15°
A7	1,04 R	0,041 R	1,04 R	0,041 R
A8	0,396± 0,010	0,0156± 0,0004	0,396± 0,010	0,0156± 0,0004
A9	0,23	0,009	0,23	0,009
A10	0,61± 0,03	0,024± 0,001	0,61± 0,03	0,024± 0,001
A11	0,23	0,009	0,23	0,009
A12	0,05	0,002	0,05	0,002
A13	5,16± 0,15	0,203± 0,006	5,16± 0,15	0,203± 0,006
A14	4,39 MAX	0,173 MAX	4,39 MAX	0,173 MAX
A15	3,02 MIN	0,119 MIN	3,02 MIN	0,119 MIN
A16	1,02± 0,25	0,040± 0,010	1,02± 0,25	0,040± 0,010
PIN X	25		34	
PIN Y	26		35	
PIN Z	50		68	

Figure 13 - 50/68-contact alternative 1/alternative 3 nonshielded mating connector
(A cable/P cable)



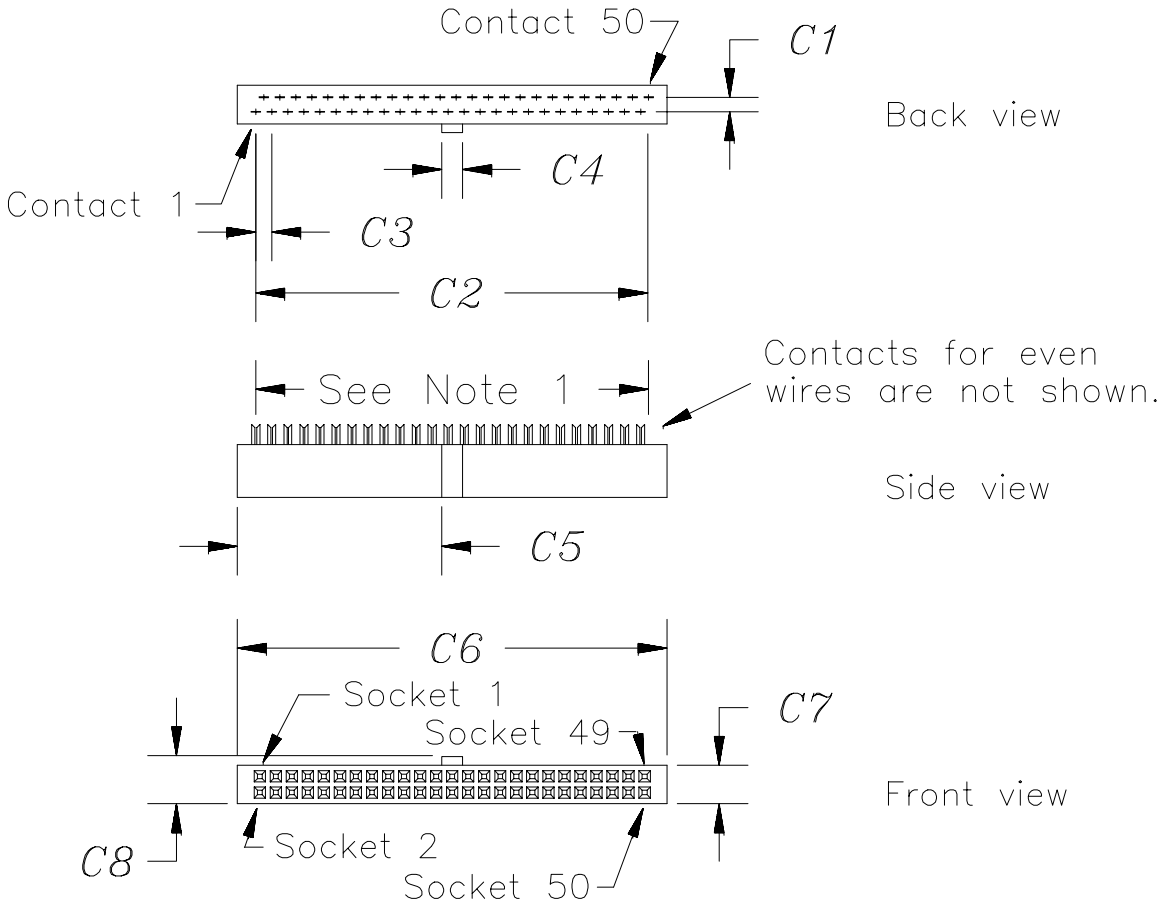
Dimension	mm	in	Comments
D1	2,54	0,100	
D2	82,80	3,260	Reference Only
D3	2,54	0,100	
D4	8,89	0,350	Reference Only
D5	72,64	2,860	Reference Only
D6	78,74	3,100	Reference Only
D7	13,94	0,549	Reference Only
D8	4,19± 0,25	0,165± 0,010	
D9	6,10	0,240	
D10	6,60	0,260	Reference Only

NOTES

1 Two rows of twenty five contacts on 2,540 mm (0,100 in) spacing = 60,960 mm (2,400 in).

2 Tolerances ±0,127 mm (0,005 in) non–cumulative, unless specified otherwise.

Figure 14 - 50-contact alternative 2 nonshielded SCSI device connector (A cable)



Dimensions	mm	in	Comments
C1	2,54	0,100	
C2	60,96	2,400	
C3	2,54	0,100	
C4	3,30	0,130	
C5	32,38	1,275	
C6	68,07	2,680	
C7	6,10	0,240	
C8	7,62	0,300	Maximum

NOTES
1 Fifty contacts on 1,270 mm (0,050 in) staggered spacing = 62,230 mm (2,450 in) [reference only].
2 Tolerances ±0,127 mm (0,005) non-cumulative, unless specified otherwise.
3 Connector cover and strain relief are optional.

Figure 15 - 50-contact alternative 2 nonshielded mating connector (A cable)

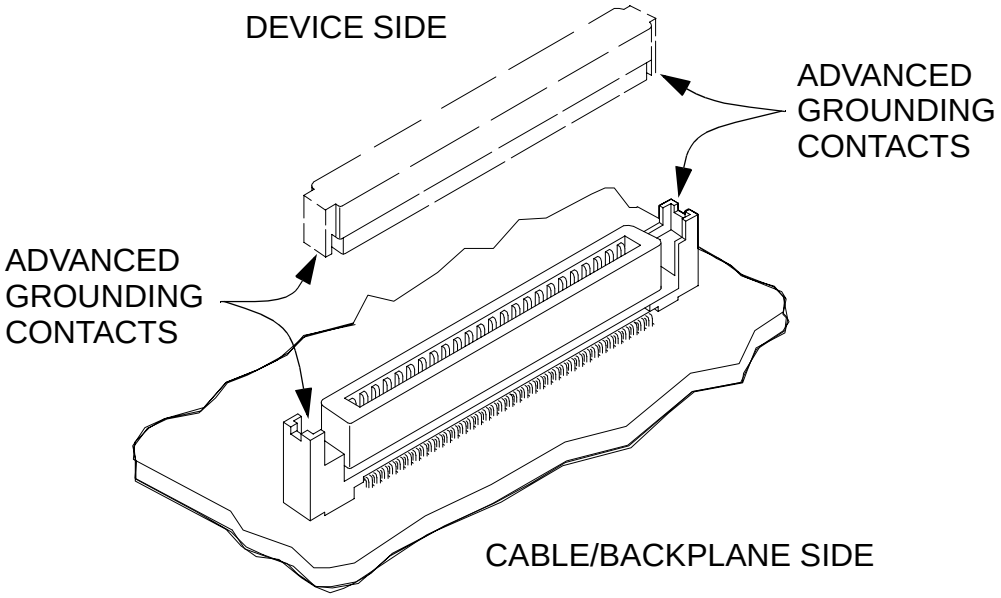


Figure 16 - 80-contact alternative 4 nonshielded SCSI device connector (P cable)

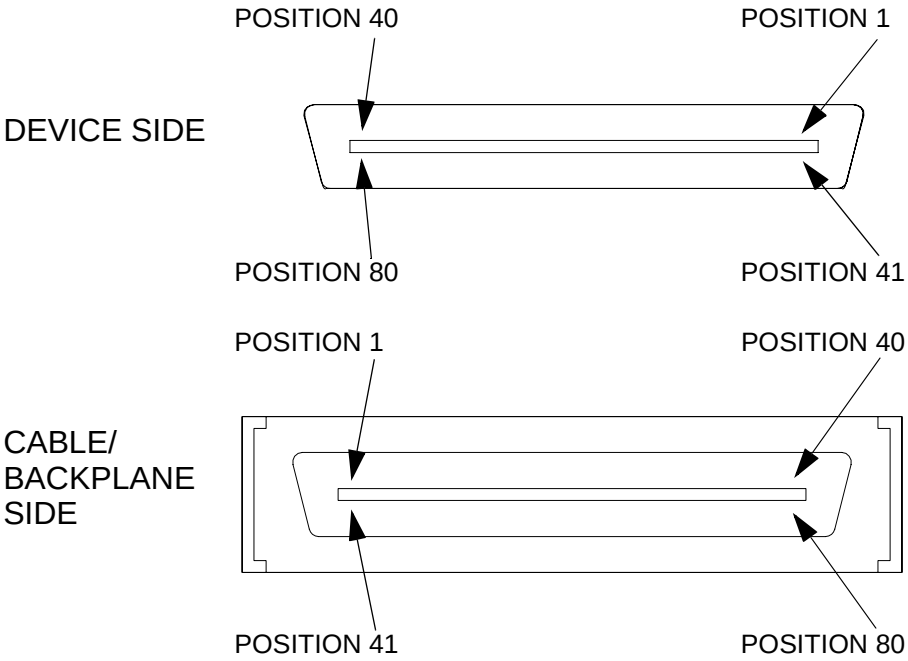


Figure 17 - 80-contact alternative 4 nonshielded contact positions (P cable)

5.3 Shielded connector

5.3.1 Shielded connector overview

Two shielded connector alternatives are specified for the A cable, and the P cable.

The D.C. resistance from the cable shield where it attaches to the connector to the enclosure should be less than 10 milliohms.

5.3.2 Shielded connector alternative 1 - A cable

The alternative 1 shielded SCSI device connector for the A cable shall be a 50-conductor connector consisting of two rows of 25 female contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 18. The nonmating portion of the connector is shown for reference only.

The alternative 1 shielded mating connector for the A cable shall be a 50-conductor connector consisting of two rows of 25 male contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 19. The nonmating portion of the connector is shown for reference only.

5.3.3 Shielded connector alternative 2 - A cable

The alternative 2 shielded SCSI device connector for the A cable shall be a 50-conductor connector consisting of two rows of ribbon contacts spaced 2,16 mm (0,085 in) apart, as shown in figure 20. The non-mating portion of the connector is shown for reference only.

The alternative 2 shielded mating connector for the A cable shall be a 50-conductor connector consisting of two rows of ribbon contacts spaced 2,16 mm (0,085 in) apart, as shown in figure 21. The non-mating portion of the connector is shown for reference only.

5.3.4 Shielded connector alternative 3 - P cable

The alternative 3 shielded SCSI device connector for the P cable shall be a 68-conductor connector consisting of two rows of 34 female contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 22. The nonmating portion of the connector is shown for reference only.

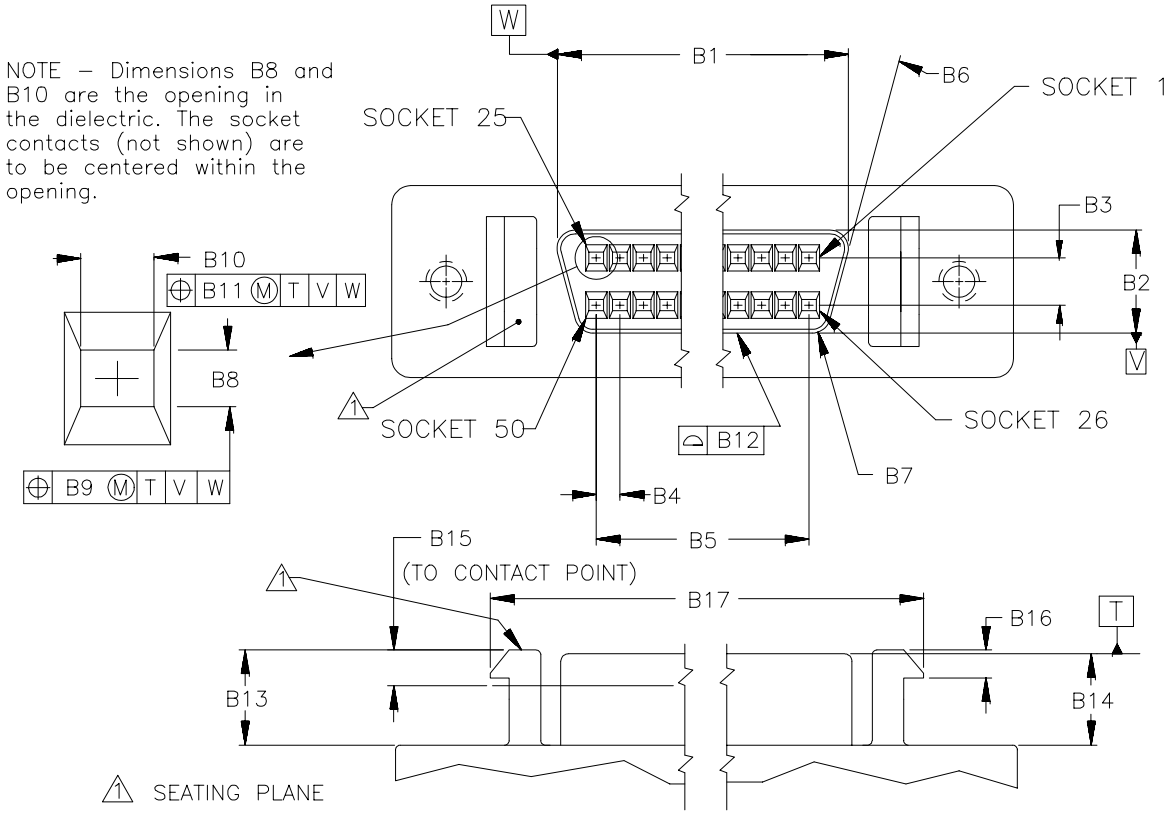
The alternative 3 shielded mating connector for the P cable shall be a 68-conductor connector consisting of two rows of 34 male contacts with adjacent contacts 1,27 mm (0,05 in) apart, as shown in figure 23. The nonmating portion of the connector is shown for reference only.

Cable retention shall consist of #2-56 thread jack screws capable of withstanding a minimum torque of 1.2 Nm (11 inch-pounds).

5.3.5 Shielded connector alternative 4 - P cable

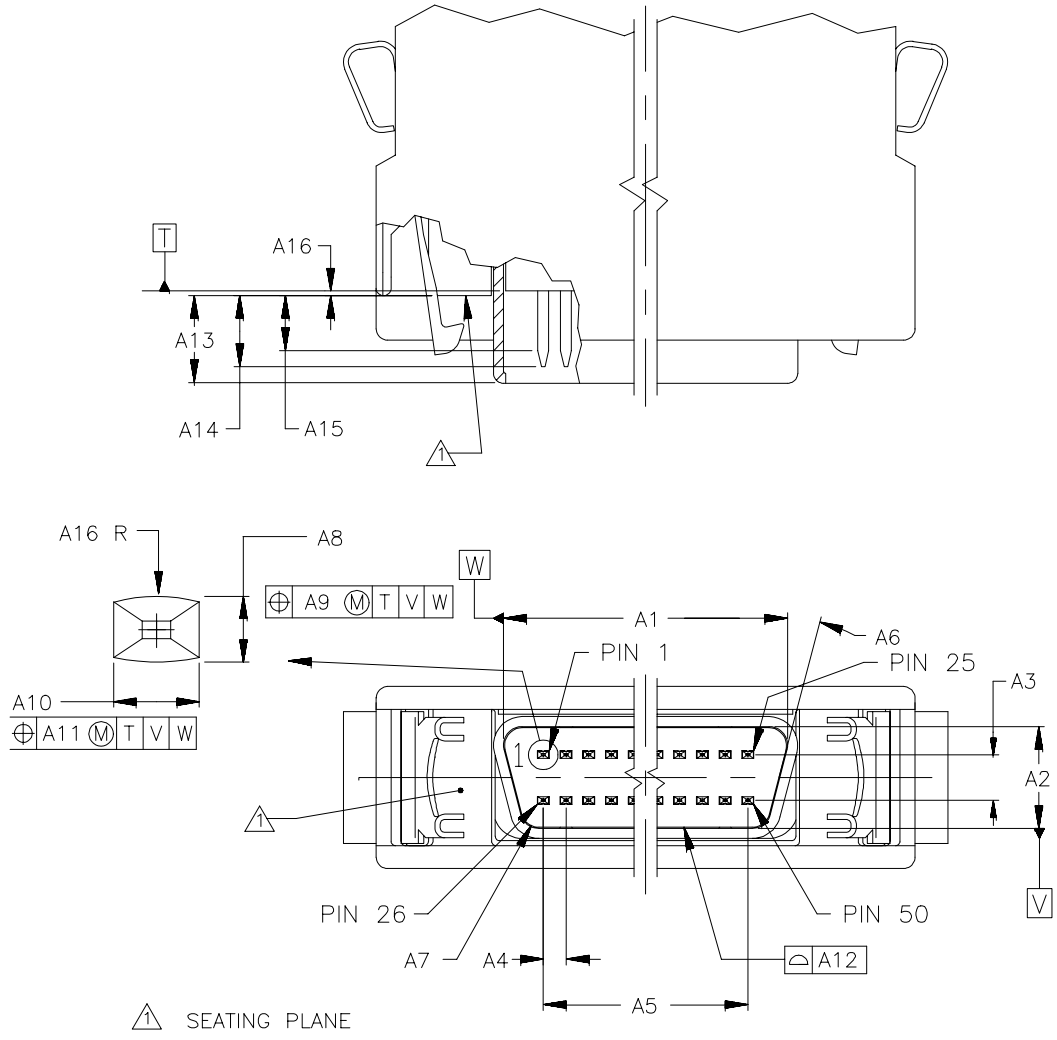
The alternative 4 shielded SCSI device connector for the P cable shall be a 68-conductor connector consisting of two rows of ribbon contacts spaced 0,8 mm (0,0315 in) apart, as shown in figure 24 and figure 25. For the detailed dimensional drawings of this connector see the VHDCI EIA standard EIA-700A0AF.

The alternative 4 shielded mating connector for the P cable shall be a 68-conductor connector consisting of two rows of ribbon contacts spaced 0,8 mm (0,0315 in) apart, as shown in figure 24 and figure 25. For the detailed dimensional drawings of this connector see the VHDCI EIA standard EIA-700A0AF and VHDCI Shielded Configurations SFF-8441.



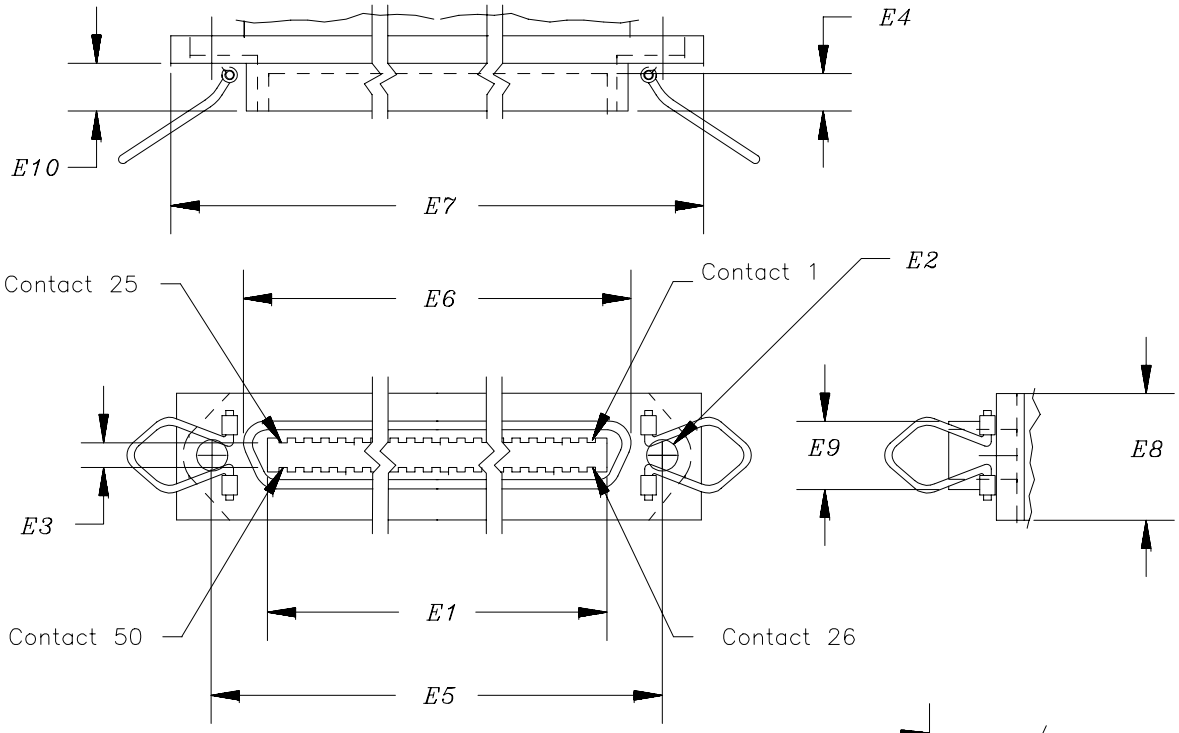
Dimensions	Millimeters	Inches
B1	34,70	1,366
B2	5,54	0,218
B3	2,54	0,100
B4	1,27	0,050
B5	30,48	1,200
B6	15°	15°
B7	1,00 R	0,039 R
B8	0,61± 0,05	0,024± 0,002
B9	0,15	0,006
B10	0,86± 0,10	0,034± 0,004
B11	0,15	0,006
B12	0,05	0,002
B13	5,10± 0,05	0,201± 0,002
B14	5,00± 0,13	0,197± 0,005
B15	1,85 max.	0,073 max.
B16	1,50± 0,03	0,059± 0,001
B17	42,29± 0,10	1,665± 0,004

Figure 18 - 50-contact alternative 1 shielded SCSI device connector (A cable)



Dimensions	Millimeters	Inches
A1	34,85	1,372
A2	5,69	0,224
A3	2,54	0,100
A4	1,27	0,050
A5	30,48	1,200
A6	15°	15°
A7	1,04 R	0,041 R
A8	0,40± 0,010	0,0156± 0,0004
A9	0,23	0,009
A10	0,60± 0,03	0,024± 0,001
A11	0,23	0,009
A12	0,05	0,002
A13	4,90± 0,10	0,193± 0,004
A14	4,27 max.	0,168 max.
A15	2,64 min.	0,104 min.
A16	0,25± 0,13	0,010± 0,005

Figure 19 - 50-contact alternative 1 shielded mating connector (A cable)



Dimensions	mm	in
E1	56,39 min.	2,220 min.
E2 1)	2,62 min.	0,103 min.
E3	3,99 min.	0,157 min.
E4	5,84 min.	0,230 min.
E5	74,85	2,947
E6	64,29 max.	2,531 max.
E7 2)	83,06	3,270
E8 2)	15,24	0,600
E9	12,04 max.	0,474 max.
E10	9,78 max.	0,385 max.
E11	2,16	0,085
E12	15° ± 2°	15° ± 2°

NOTE – Tolerances ± 0,127 mm (0,005 in) non–cumulative, unless specified otherwise.

1) This dimension is selected to accomodate 4–40 or 6–32 threaded screws.

2) These dimensions are shown for reference only.

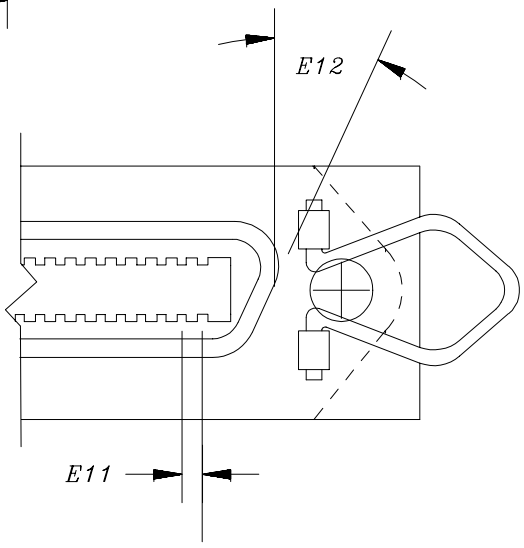
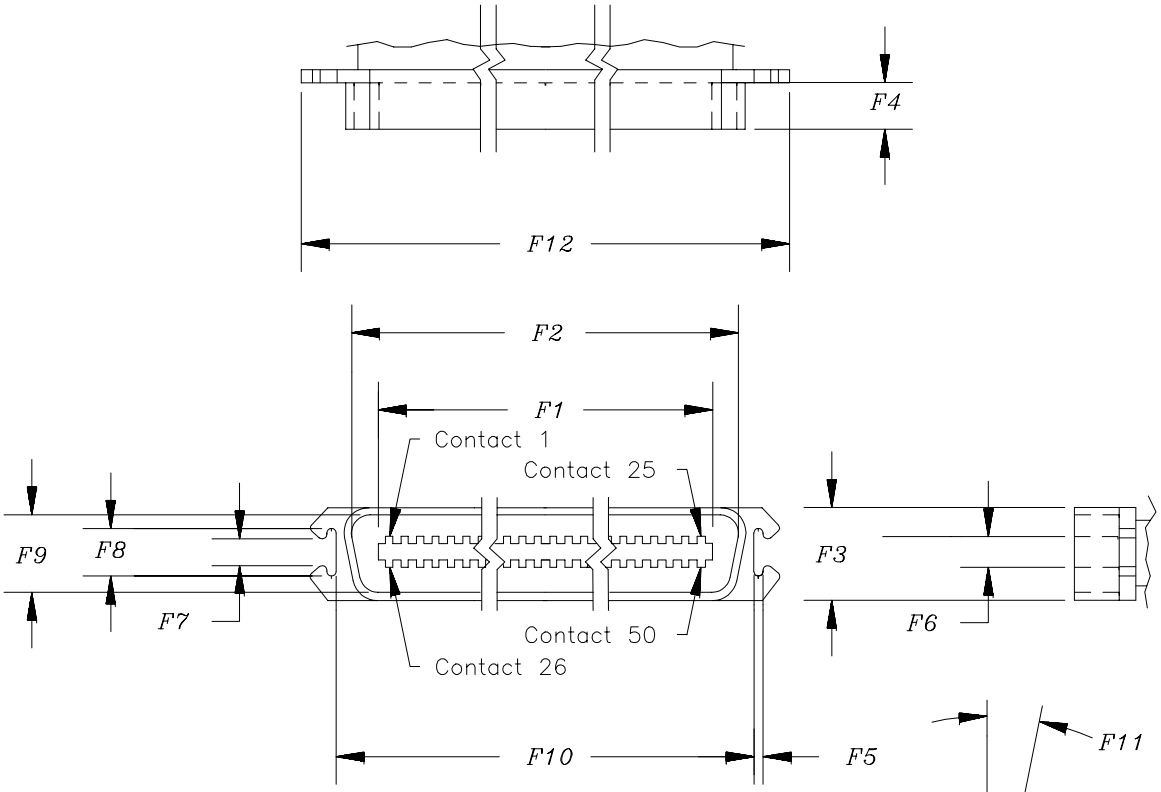


Figure 20 - 50-contact alternative 2 shielded SCSI device connector (A cable)

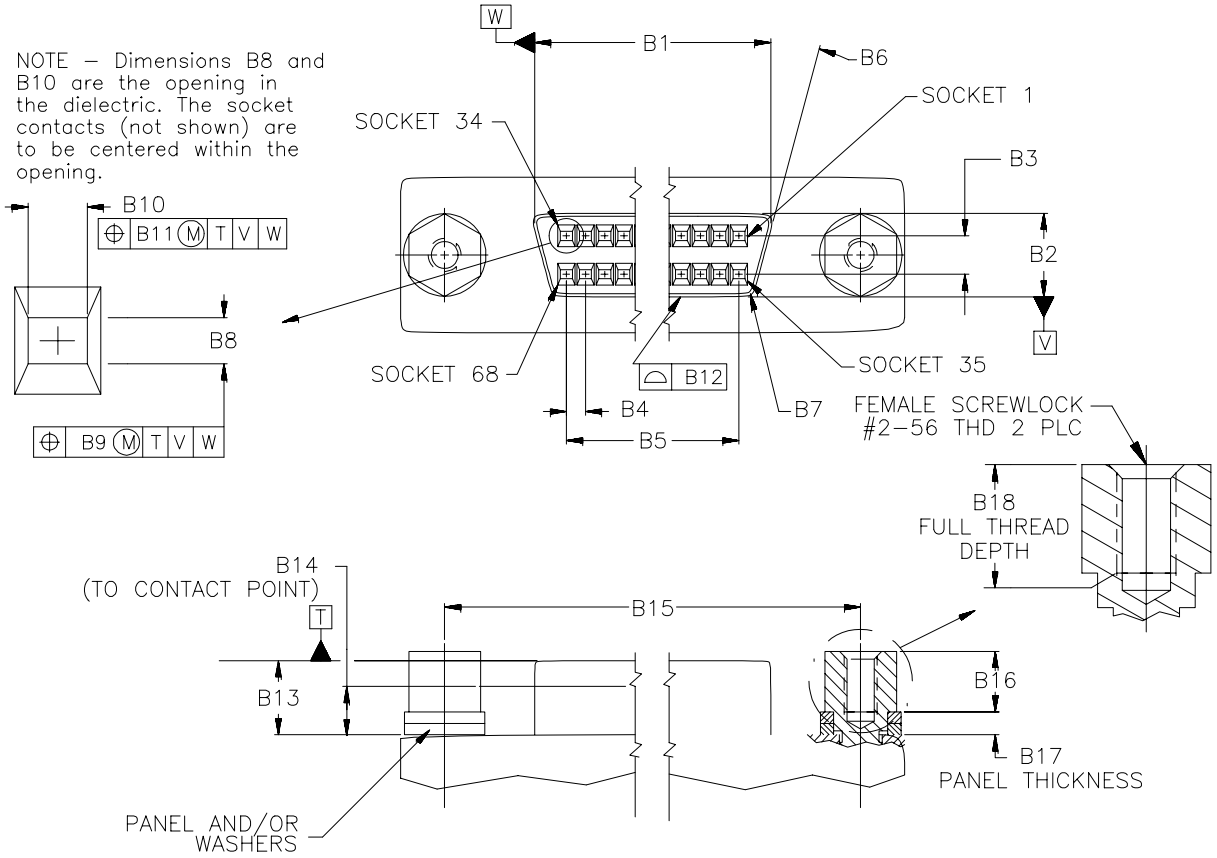


Dimensions	mm	in
F1	56,26 max.	2,215 max.
F2	64,29 min.	2,531 min.
F3 1)	15,24	0,600
F4	7,29 min.	0,287 min.
F5 1)	1,02	0,040
F6	4,09 max.	0,161 max.
F7 1)	5,08	0,200
F8 1)	6,10	0,240
F9	12,04 min.	0,474 min.
F10 1)	68,45	2,695
F11	15° ± 2°	15° ± 2°
F12 1)	76,71	3,020
F13	2,16	0,085

NOTE – Tolerances ± 0,127 mm (0,005 in) non–cumulative, unless specified otherwise.

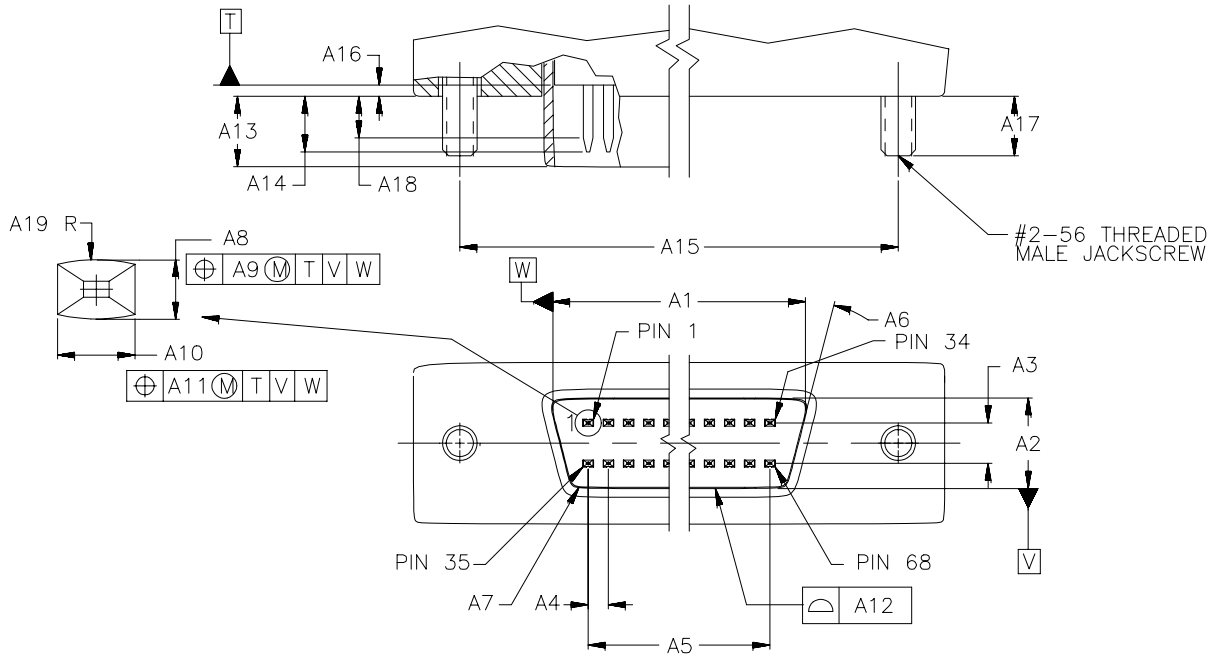
1) These dimensions are shown for reference only.

Figure 21 - 50-contact alternative 2 shielded mating connector (A cable)



Dimensions	Millimeters	Inches
B1	46,13	1,816
B2	5,54	0,218
B3	2,54	0,100
B4	1,27	0,050
B5	41,91	1,650
B6	15°	15°
B7	1,00 R	0,039 R
B8	0,61± 0,05	0,024± 0,002
B9	0,15	0,006
B10	0,86± 0,10	0,034± 0,004
B11	0,15	0,006
B12	0,05	0,002
B13	5,00± 0,13	0,197± 0,005
B14	3,30 min.	0,130 min.
B15	57,91± 0,13	2,280± 0,005
B16	3,71± 0,05	0,146± 0,002
B17	1,52± 0.13	0,060± 0,005
B18	3,58 min.	0,141 min.

Figure 22 - 68-contact alternative 3 shielded SCSI device connector (P cable)



Dimensions	Millimeters	Inches
A1	46,28	1,822
A2	5,69	0,224
A3	2,54	0,100
A4	1,27	0,050
A5	41,91	1,650
A6	15°	15°
A7	1,04 R	0,041 R
A8	0,396± 0,010	0,0156± 0,0004
A9	0,23	0,009
A10	0,60± 0,03	0,024± 0,001
A11	0,23	0,009
A12	0,05	0,002
A13	4,90± 0,10	0,193± 0,004
A14	4,27 max.	0,168 max.
A15	57,91± 0,13	2,280± 0,005
A16	0,25± 0,13	0,010± 0,005
A17	3,43± 0,15	0,135± 0,006
A18	2,64 min.	0,104 min.
A19	1,02± 0,25	0,040± 0,010

Figure 23 - 68-contact alternative 3 shielded mating connector (P cable)

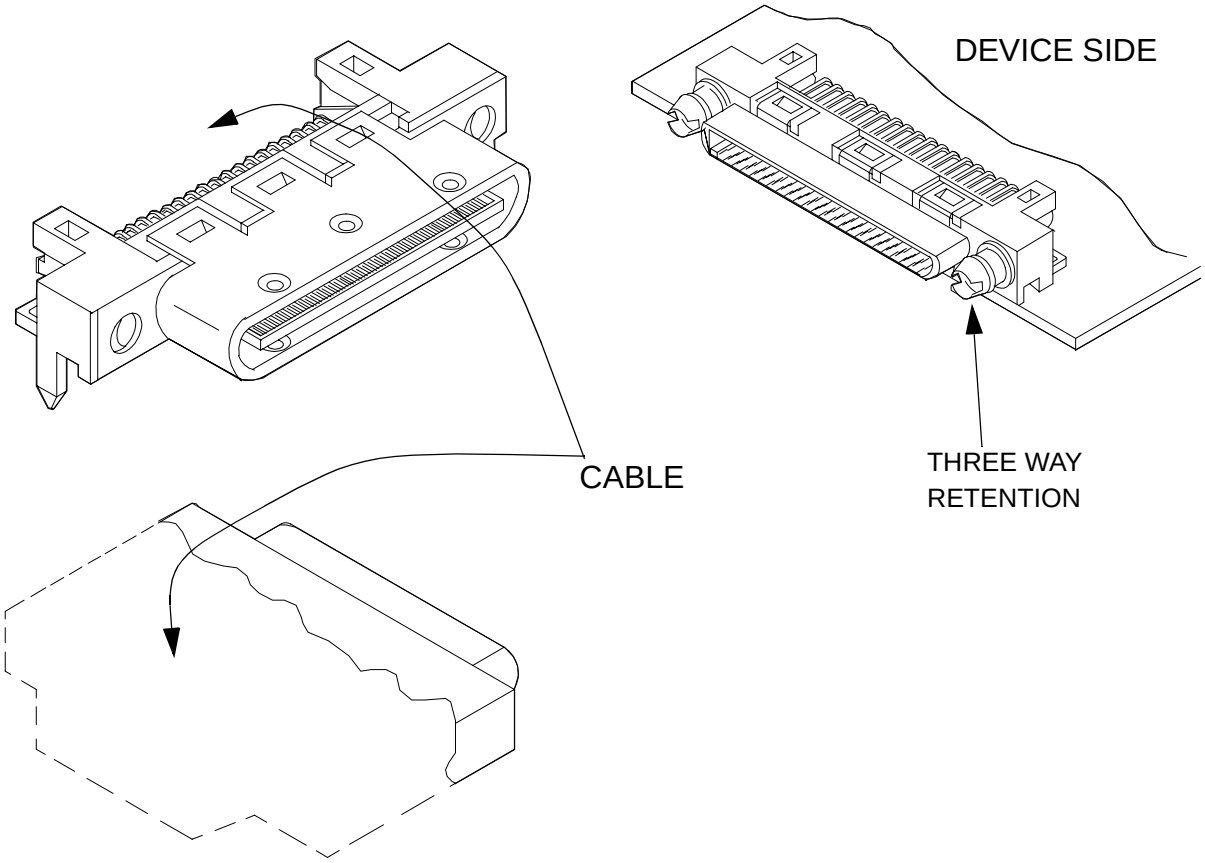


Figure 24 - 68-contact alternative 4 shielded SCSI device connector (P cable)

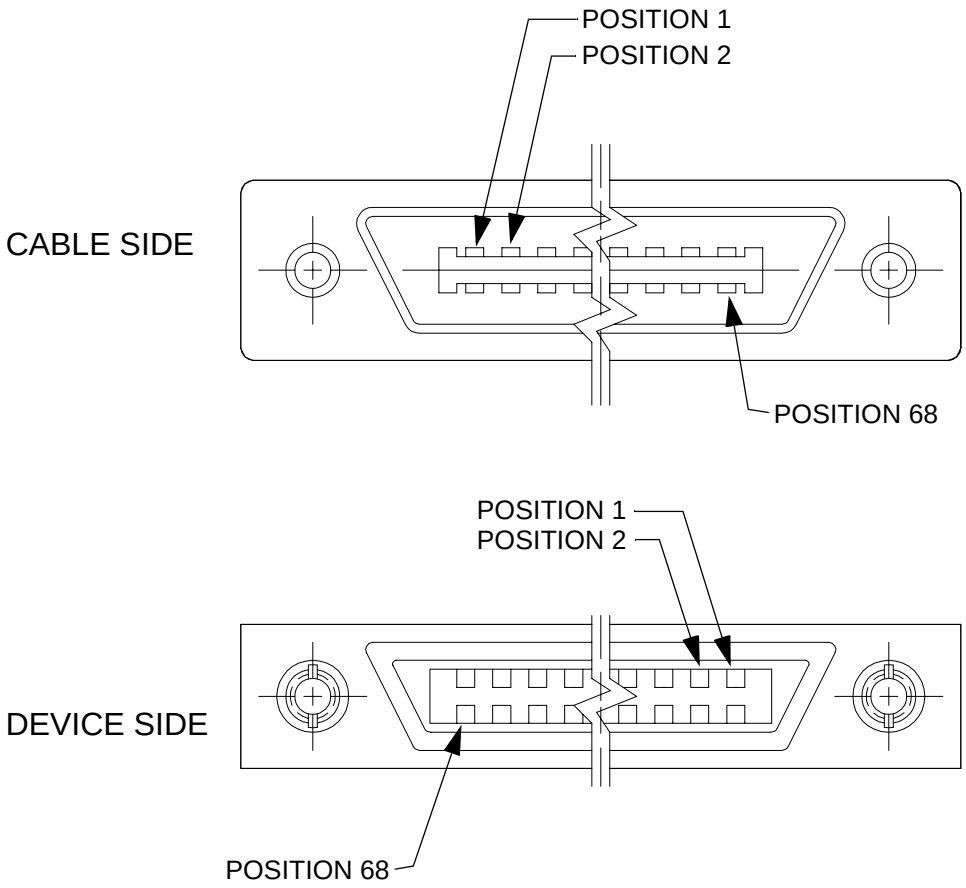


Figure 25 - 68-contact alternative 4 shielded contact positions (P cable)

5.4 Connector contact assignments

5.4.1 Connector contact assignments overview

The connector contact assignments are defined in tables 2 through 8. See 8.2 for the definitions of the signals. The items under signal name labelled TERMPWR, and RESERVED are not signals and are not required to meet the cable characteristics for signals in 6.3. See 6.4 for characteristics of TERMPWR. See 6.5 for characteristics of RESERVED lines.

Table 2 - Cross-reference to A cable contact assignments

Connector type	Transmission mode	Connector figure	Contact assignment table	Contact set
Nonshielded alternative 1	SE	12 and 13	3	2
Nonshielded alternative 1	LVD	12 and 13	6	2
Nonshielded alternative 2	SE	14 and 15	3	1
Nonshielded alternative 2	LVD	14 and 15	6	1
Shielded alternative 1	SE	18 and 19	3	2
Shielded alternative 1	LVD	18 and 19	6	2
Shielded alternative 2	SE	20 and 21	3	1
Shielded alternative 2	LVD	20 and 21	6	1

5.4.2 SE assignments

Table 3 defines the connector contact assignments for a 50 conductor bus that uses SE transceivers.

Table 3 - SE contact assignments - A cable

Signal name	Connector contact number		Cable conductor number		Connector contact number		Signal name
	Set 2	Set 1			Set 1	Set 2	
SIGNAL RETURN	1	1	1	2	2	26	-DB(0)
SIGNAL RETURN	2	3	3	4	4	27	-DB(1)
SIGNAL RETURN	3	5	5	6	6	28	-DB(2)
SIGNAL RETURN	4	7	7	8	8	29	-DB(3)
SIGNAL RETURN	5	9	9	10	10	30	-DB(4)
SIGNAL RETURN	6	11	11	12	12	31	-DB(5)
SIGNAL RETURN	7	13	13	14	14	32	-DB(6)
SIGNAL RETURN	8	15	15	16	16	33	-DB(7)
SIGNAL RETURN	9	17	17	18	18	34	-P_CRCA
GROUND	10	19	19	20	20	35	GROUND
GROUND	11	21	21	22	22	36	GROUND
RESERVED	12	23	23	24	24	37	RESERVED
OPEN (1)	13	25	25	26	26	38	TERMPWR
RESERVED	14	27	27	28	28	39	RESERVED
GROUND	15	29	29	30	30	40	GROUND
SIGNAL RETURN	16	31	31	32	32	41	-ATN
GROUND	17	33	33	34	34	42	GROUND
SIGNAL RETURN	18	35	35	36	36	43	-BSY
SIGNAL RETURN	19	37	37	38	38	44	-ACK
SIGNAL RETURN	20	39	39	40	40	45	-RST
SIGNAL RETURN	21	41	41	42	42	46	-MSG
SIGNAL RETURN	22	43	43	44	44	47	-SEL
SIGNAL RETURN	23	45	45	46	46	48	-C/D
SIGNAL RETURN	24	47	47	48	48	49	-REQ
SIGNAL RETURN	25	49	49	50	50	50	-I/O

Notes

1 Open lines shall be open in all SCSI devices and terminators.

2 The minus sign next to a signal indicates asserted low.

3 The conductor number refers to the conductor position when using planar cable media.

4 Two sets of contact assignments are shown, Refer to table 2 to determine which set of contacts applies to each connector.

Table 4 defines the connector contact assignments for a 68 conductor bus that uses SE transceivers.

Table 4 - SE contact assignments - P cable

Signal name	Connector contact number	Cable conductor number		Connector contact number	Signal name
SIGNAL RETURN	1	1	2	35	-DB(12)
SIGNAL RETURN	2	3	4	36	-DB(13)
SIGNAL RETURN	3	5	6	37	-DB(14)
SIGNAL RETURN	4	7	8	38	-DB(15)
SIGNAL RETURN	5	9	10	39	-DB(P1)
SIGNAL RETURN	6	11	12	40	-DB(0)
SIGNAL RETURN	7	13	14	41	-DB(1)
SIGNAL RETURN	8	15	16	42	-DB(2)
SIGNAL RETURN	9	17	18	43	-DB(3)
SIGNAL RETURN	10	19	20	44	-DB(4)
SIGNAL RETURN	11	21	22	45	-DB(5)
SIGNAL RETURN	12	23	24	46	-DB(6)
SIGNAL RETURN	13	25	26	47	-DB(7)
SIGNAL RETURN	14	27	28	48	-P_CRCA
GROUND	15	29	30	49	GROUND
GROUND	16	31	32	50	GROUND
TERMPWR	17	33	34	51	TERMPWR
TERMPWR	18	35	36	52	TERMPWR
RESERVED	19	37	38	53	RESERVED
GROUND	20	39	40	54	GROUND
SIGNAL RETURN	21	41	42	55	-ATN
GROUND	22	43	44	56	GROUND
SIGNAL RETURN	23	45	46	57	-BSY
SIGNAL RETURN	24	47	48	58	-ACK
SIGNAL RETURN	25	49	50	59	-RST
SIGNAL RETURN	26	51	52	60	-MSG
SIGNAL RETURN	27	53	54	61	-SEL
SIGNAL RETURN	28	55	56	62	-C/D
SIGNAL RETURN	29	57	58	63	-REQ
SIGNAL RETURN	30	59	60	64	-I/O
SIGNAL RETURN	31	61	62	65	-DB(8)
SIGNAL RETURN	32	63	64	66	-DB(9)
SIGNAL RETURN	33	65	66	67	-DB(10)
SIGNAL RETURN	34	67	68	68	-DB(11)

Notes
1 The minus sign next to a signal indicates asserted low.
2 The conductor number refers to the conductor position when using planar cable media.

Table 5 defines the connector contact assignments for an 80 conductor bus that uses SE transceivers.

Table 5 - SE contact assignments - nonshielded alternative 4 connector

Signal name	Host Pin Length (note 3)	Connector contact number	Connector contact number	Host Pin Length (note 3)	Signal name
12V CHARGE (note 1)	Long	1	41	Long	12V GROUND (note 1)
12V (note 1)	Short	2	42	Long	12V GROUND (note 1)
12V (note 1)	Short	3	43	Long	12V GROUND (note 1)
12V (note 1)	Short	4	44	Short	MATED 1 (note 1)
3,3V (note 1)	Short	5	45	Long	3,3V CHARGE (note 1)
3,3V (note 1)	Short	6	46	Long	GROUND
-DB(11)	Short	7	47	Short	SIGNAL RETURN
-DB(10)	Short	8	48	Short	SIGNAL RETURN
-DB(9)	Short	9	49	Short	SIGNAL RETURN
-DB(8)	Short	10	50	Short	SIGNAL RETURN
-I/O	Short	11	51	Short	SIGNAL RETURN
-REQ	Short	12	52	Short	SIGNAL RETURN
-C/D	Short	13	53	Short	SIGNAL RETURN
-SEL	Short	14	54	Short	SIGNAL RETURN
-MSG	Short	15	55	Short	SIGNAL RETURN
-RST	Short	16	56	Short	SIGNAL RETURN
-ACK	Short	17	57	Short	SIGNAL RETURN
-BSY	Short	18	58	Short	SIGNAL RETURN
-ATN	Short	19	59	Short	SIGNAL RETURN
-P_CRCA	Short	20	60	Short	SIGNAL RETURN
-DB(7)	Short	21	61	Short	SIGNAL RETURN
-DB(6)	Short	22	62	Short	SIGNAL RETURN
-DB(5)	Short	23	63	Short	SIGNAL RETURN
-DB(4)	Short	24	64	Short	SIGNAL RETURN
-DB(3)	Short	25	65	Short	SIGNAL RETURN
-DB(2)	Short	26	66	Short	SIGNAL RETURN
-DB(1)	Short	27	67	Short	SIGNAL RETURN
-DB(0)	Short	28	68	Short	SIGNAL RETURN
-DB(P1)	Short	29	69	Short	SIGNAL RETURN
-DB(15)	Short	30	70	Short	SIGNAL RETURN
-DB(14)	Short	31	71	Short	SIGNAL RETURN
-DB(13)	Short	32	72	Short	SIGNAL RETURN
-DB(12)	Short	33	73	Short	SIGNAL RETURN
5V (note 1)	Short	34	74	Short	MATED 2 (note 1)
5V (note 1)	Short	35	75	Long	5V GROUND (note 1)
5V CHARGE (note 1)	Long	36	76	Long	5V GROUND (note 1)
SPINDLE SYNC (note 1)	Long	37	77	Long	ACTIVE LED OUT (note 1)
RMT_START (note 1)	Long	38	78	Long	DLYD_START (note 1)
SCSI ID (0) (note 1)	Long	39	79	Long	SCSI ID (1) (note 1)
SCSI ID (2) (note 1)	Long	40	80	Long	SCSI ID (3) (note 1)

Notes

1 See Annex C for the definition of these signals.

2 The minus sign next to a signal indicates asserted low.

3 The pins identified as being short and long only applies to the cable/backplane side connector and not the SCSI device side connector. All pins on the SCSI device side connector are the same length.

5.4.3 Differential connector contact assignments

Table 6 defines the connector contact assignments for a 50 conductor bus that uses LVD and LVD/MSE transceivers.

Table 6 - LVD/MSE contact assignments - A cable

Signal name	Connector contact number		Cable conductor number		Connector contact number		Signal name
	Set 2	Set 1			Set 1	Set 2	
+DB(0)	1	1	1	2	2	26	-DB(0)
+DB(1)	2	3	3	4	4	27	-DB(1)
+DB(2)	3	5	5	6	6	28	-DB(2)
+DB(3)	4	7	7	8	8	29	-DB(3)
+DB(4)	5	9	9	10	10	30	-DB(4)
+DB(5)	6	11	11	12	12	31	-DB(5)
+DB(6)	7	13	13	14	14	32	-DB(6)
+DB(7)	8	15	15	16	16	33	-DB(7)
+P_CRCA	9	17	17	18	18	34	-P_CRCA
GROUND	10	19	19	20	20	35	GROUND
DIFFSENS	11	21	21	22	22	36	GROUND
RESERVED	12	23	23	24	24	37	RESERVED
TERMPWR	13	25	25	26	26	38	TERMPWR
RESERVED	14	27	27	28	28	39	RESERVED
GROUND	15	29	29	30	30	40	GROUND
+ATN	16	31	31	32	32	41	-ATN
GROUND	17	33	33	34	34	42	GROUND
+BSY	18	35	35	36	36	43	-BSY
+ACK	19	37	37	38	38	44	-ACK
+RST	20	39	39	40	40	45	-RST
+MSG	21	41	41	42	42	46	-MSG
+SEL	22	43	43	44	44	47	-SEL
+C/D	23	45	45	46	46	48	-C/D
+REQ	24	47	47	48	48	49	-REQ
+I/O	25	49	49	50	50	50	-I/O

Notes

1 The conductor number refers to the conductor position when using planar cable media.

2 Two sets of contact assignments are shown, Refer to table 2 to determine which set of contacts applies to each connector.

Table 7 defines the connector contact assignments for a 68 conductor bus that uses LVD and LVD/MSE transceivers.

Table 7 - LVD/MSE contact assignments - P cable

Signal name	Connector contact number	Cable conductor number		Connector contact number	Signal name
+DB(12)	1	1	2	35	-DB(12)
+DB(13)	2	3	4	36	-DB(13)
+DB(14)	3	5	6	37	-DB(14)
+DB(15)	4	7	8	38	-DB(15)
+DB(P1)	5	9	10	39	-DB(P1)
+DB(0)	6	11	12	40	-DB(0)
+DB(1)	7	13	14	41	-DB(1)
+DB(2)	8	15	16	42	-DB(2)
+DB(3)	9	17	18	43	-DB(3)
+DB(4)	10	19	20	44	-DB(4)
+DB(5)	11	21	22	45	-DB(5)
+DB(6)	12	23	24	46	-DB(6)
+DB(7)	13	25	26	47	-DB(7)
+P_CRCA	14	27	28	48	-P_CRCA
GROUND	15	29	30	49	GROUND
DIFFSENS	16	31	32	50	GROUND
TERMPWR	17	33	34	51	TERMPWR
TERMPWR	18	35	36	52	TERMPWR
RESERVED	19	37	38	53	RESERVED
GROUND	20	39	40	54	GROUND
+ATN	21	41	42	55	-ATN
GROUND	22	43	44	56	GROUND
+BSY	23	45	46	57	-BSY
+ACK	24	47	48	58	-ACK
+RST	25	49	50	59	-RST
+MSG	26	51	52	60	-MSG
+SEL	27	53	54	61	-SEL
+C/D	28	55	56	62	-C/D
+REQ	29	57	58	63	-REQ
+I/O	30	59	60	64	-I/O
+DB(8)	31	61	62	65	-DB(8)
+DB(9)	32	63	64	66	-DB(9)
+DB(10)	33	65	66	67	-DB(10)
+DB(11)	34	67	68	68	-DB(11)

Notes

1 The conductor number refers to the conductor position when using planar cable media.

Table 8 defines the connector contact assignments for an 80 conductor bus that uses LVD and LVD/MSE transceivers.

Table 8 - LVD/MSE contact assignments - nonshielded alternative 4 connector

Signal name	Host Pin Length (note 2)	Connector contact number	Connector contact number	Host Pin Length (note 2)	Signal name
12V CHARGE (note 1)	Long	1	41	Long	12V GROUND (note 1)
12V (note 1)	Short	2	42	Long	12V GROUND (note 1)
12V (note 1)	Short	3	43	Long	12V GROUND (note 1)
12V (note 1)	Short	4	44	Short	MATED 1 (note 1)
3,3V (note 1)	Short	5	45	Long	3,3V CHARGE (note 1)
3,3V (note 1)	Short	6	46	Long	DIFFSNS
-DB(11)	Short	7	47	Short	+DB(11)
-DB(10)	Short	8	48	Short	+DB(10)
-DB(9)	Short	9	49	Short	+DB(9)
-DB(8)	Short	10	50	Short	+DB(8)
-I/O	Short	11	51	Short	+I/O
-REQ	Short	12	52	Short	+REQ
-C/D	Short	13	53	Short	+C/D
-SEL	Short	14	54	Short	+SEL
-MSG	Short	15	55	Short	+MSG
-RST	Short	16	56	Short	+RST
-ACK	Short	17	57	Short	+ACK
-BSY	Short	18	58	Short	+BSY
-ATN	Short	19	59	Short	+ATN
-P_CRCA	Short	20	60	Short	+P_CRCA
-DB(7)	Short	21	61	Short	+DB(7)
-DB(6)	Short	22	62	Short	+DB(6)
-DB(5)	Short	23	63	Short	+DB(5)
-DB(4)	Short	24	64	Short	+DB(4)
-DB(3)	Short	25	65	Short	+DB(3)
-DB(2)	Short	26	66	Short	+DB(2)
-DB(1)	Short	27	67	Short	+DB(1)
-DB(0)	Short	28	68	Short	+DB(0)
-DB(P1)	Short	29	69	Short	+DB(P1)
-DB(15)	Short	30	70	Short	+DB(15)
-DB(14)	Short	31	71	Short	+DB(14)
-DB(13)	Short	32	72	Short	+DB(13)
-DB(12)	Short	33	73	Short	+DB(12)
5V (note 1)	Short	34	74	Short	MATED 2 (note 1)
5V (note 1)	Short	35	75	Long	5V GROUND (note 1)
5V CHARGE (note 1)	Long	36	76	Long	5V GROUND (note 1)
SPINDLE SYNC (note 1)	Long	37	77	Long	ACTIVE LED OUT (note 1)
RMT_START (note 1)	Long	38	78	Long	DLYD_START (note 1)
SCSI ID (0) (note 1)	Long	39	79	Long	SCSI ID (1) (note 1)
SCSI ID (2) (note 1)	Long	40	80	Long	SCSI ID (3) (note 1)

Notes

1 See Annex C for the definition of these signals.

2 The pins identified as being short and long only applies to the cable/backplane side connector and not the SCSI device side connector. All pins on the SCSI device side connector are the same length.

6 SCSI bus interconnect

6.1 SCSI bus interconnect overview

This clause defines the characteristics of interconnects used to connect SPI devices. These interconnects are part of the SCSI path. See figure 4 for the topology of the interconnects that make up a SCSI path. Examples of types of Interconnects are:

a) unshielded planar cable media;
b) unshielded round twisted-pair cables;
c) shielded round twisted-pair cables;
d) backplanes.

The interconnect is defined as the electrical media (including connectors and passive loads) used to connect the TERMPWR, terminators, and SCSI devices in a SCSI bus.

The functions of the interconnects are to:

a) carry the signals,
b) carry the terminator power from TERMPWR sources to the terminators, and
c) to provide continuity between reserved pins and ground pins between SCSI devices and terminators.

The interconnect shall meet the specified characteristics to ensure that compliant worst case transmitted signals result in received signals that meet the requirements in clause 7. Signals for this requirement include DB(0) through DB(15), P_CRCA, DB(P1), C/D, I/O, MSG, BSY, SEL, ATN, REQ, ACK, DIFFSENS, and RESET. At least minimum TERMPWR shall be delivered to the terminator from minimum sources per the requirements in subclause 7.5.

The requirements on interconnect components in this clause are intended to produce interchangeable components while achieving the required signal transmission properties.

6.2 SCSI bus cables

If twisted-pair cables are used, the twisted pairs in the cable shall be wired to physically opposing contacts in the connector.

The following requirements based on the connector contact assignments in 5.4.2 ensure that all SCSI round cables may be used with either SE, MSE, or LVD transceivers:

a) In the P cable conductor pairs #47-48 (ACK) and #57-58 (REQ) shall be in the cable core;
b) In the P cable, if there are more than four conductor pairs in the cable core, conductor pairs #47-48 (ACK) and #57-58 (REQ) shall not be adjacent to each other;
c) In the A cable conductor pairs #37-38 (ACK) and #47-48 (REQ) shall be in the cable core;
d) In the A cable, if there are more than three conductor pairs in the cable core, conductor pairs #37-38 (ACK) and #47-48 (REQ) shall not be adjacent to each other;
e) Cable conductor pairs used for the DATA BUS (DBnP1) and P_CRCA shall be in the outer layer of the cable;
f) Each cable conductor pair shall consist of the signal return and its associated signal for single ended applications or signal minus and its associated signal plus for differential applications.

Crosstalk noise is minimized by conductor placement (REQ and ACK in the center, data around the periphery) in round, twisted-pair cables and by the pin assignments on the connector on planar cables.

See Annex H for information on interconnecting busses of different widths.

The items under the signal name labelled TERMPWR are not signals and are not required to meet the cable characteristics for signals in 6.3. See 6.4 for characteristics of TERMPWR.

See 6.5 for characteristics of RESERVED lines.

Interconnection of SCSI devices by means other than cables is allowed (e.g., by backplanes using printed wiring boards) (see Annex L). Detailed descriptions of these other means are not part of this standard; however, they are subject to the electrical requirements in clause 6.

A SCSI bus carries an 8-bit or 16-bit DATA BUS and the signals used to move information between SCSI devices.

The signals shall not be internally connected together within the connectors or cables. See 8.2 for signal definitions.

6.3 Interconnect characteristics for signals

6.3.1 Applicability of requirements for SCSI cable media

The requirements in this clause apply to uniform cable media. Non-uniform media is media that contains dissimilar sections for purposes of enabling connector attachment.

Non-uniformities (e.g., a planar section created for connector attachment) are considered to be part of a cable assembly or harness whose performance is affected by the attached (sometimes unused) connectors as well as by the non-uniformity in the media.

Implementors using non-uniform media may construct special uniform test media using manufacturing processes similar to that used for the non-uniform media for purposes of measuring the properties of the media between the connector attachment areas (e.g. the twisted regions in a twisted/straight planar construction).

For length dependent parameters both total and per unit length requirements are specified. This ensures performance compliance when concatenating cables in the same SCSI bus segment. Implementors have the practical option to use only the total requirements and to loosen the per unit length requirements in non-concatenated applications; however, this practice creates non-conforming cables. Any cable media not meeting the per unit length requirements shall be labeled in a manner indicating that it is not suitable for use in cable assemblies that may be used in a concatenated manner.

Meeting the SCSI signal requirements in complete SCSI segments may require allowances beyond the uniform media requirements specified in this clause. See Annex E.

The requirements in this clause apply to all the SCSI signals in the cable except where otherwise specified.

6.3.2 Minimum conductor size for signals

The minimum conductor size for signals should be as specified in table 11.

6.3.3 Local transmission line impedance

The SE transmission line impedance of the cable and the differential transmission line impedance of the cable is defined in table 9.

Table 9 - SE and LVD local transmission line impedance

Cable construction	Local SE transmission line impedance (note 2)		Local differential transmission line impedance (note 2)	
	Minimum	Maximum	Minimum	Maximum
All	84 Ohms (78 Ohms) (note 3)	96 Ohms	110 Ohms	135 Ohms

Note:
1 - All values are measured by time domain reflectometry
2 - Ideally one design will meet both SE and differential criteria
3 - If SCSI loads attached to the cable media are separated by more than 1,0 m use the value of 78 Ohms.

6.3.4 Extended distance transmission line impedance

The swept frequency (extended distance) differential impedance limits shall be a maximum peak to peak variation of 60 Ohms over the frequency range 30 MHz to 600 MHz on a 30 meter cable.

6.3.5 Capacitance

The capacitance limits for cable media are as shown in table 10.

Table 10 - Cable media capacitance limits

	Minimum capacitance	Maximum capacitance	Frequency
SE	30 pF/m	66 pF/m	100 kHz and 1 MHz
Differential	26 pF/m	46 pF/m	100 kHz and 1 MHz

The dielectric constant variation in the material forming the insulation directly in contact with the conductors in the cable media between 300 kHz and 600 MHz shall be the less than 1,10 when the maximum dielectric constant in the frequency range divided by the minimum dielectric constant in the frequency range.

6.3.6 SE propagation time and propagation time skew

SE propagation time requirements are not separately specified. SE propagation time requirements are indirectly specified through the differential requirements in 6.3.7.

6.3.7 Differential propagation time and propagation time skew

The differential propagation time shall be no greater than 5,4 ns/m within the cable media and no greater than 135 ns from termination to terminator.

The differential propagation time skew (pair to pair) shall be no greater than 82 ps/m within the cable media and no greater than 2,0 ns from terminator to terminator.

6.3.8 SE attenuation

SE attenuation requirements are not separately specified. SE attenuation requirements are indirectly specified through the differential requirements in 6.3.9.

6.3.9 Differential attenuation

The attenuation requirements for differential attenuation are specified in table 11.

Both the per meter and the length equivalent to the terminator to terminator spacing requirements in table 11 shall be simultaneously met.

Table 11 - Attenuation requirements for SCSI cable media

Distance between SCSI bus terminators (meters)	Attenuation per meter maximum (dB) at 200 MHz	Attenuation of length equivalent to terminator to terminator distance maximum (dB) at 200 MHz	Distances are consistent with these minimum size conductors when used with high quality dielectrics	Notes
0 to 9	0,63	6	0,032 4 mm^2 (32 AWG) solid/ 0,050 92 mm^2 (30 AWG) stranded	multiple loads allowed
0 to 12	0,48	6	0,050 92 mm^2 (30 AWG) solid/ 0,080 42 mm^2 (28 AWG) stranded	multiple loads allowed
>12 to 25	0,48	12	0,050 92 mm^2 (30 AWG) solid/ 0,080 42 mm^2 (28 AWG) stranded	point to point only
Note: Both the per meter and the length equivalent to the terminator to terminator spacing requirements shall be simultaneously met				

6.3.10 Crosstalk

The maximum NEXT on REQ or ACK is 3,0 % of the 1,0 ns rise time aggressor signal amplitude. Crosstalk percent is calculated as follows:

$$\%NEXT = \frac{\Sigma \text{ peak absolute differential induced voltages on REQ or ACK}}{\text{peak to peak differential aggressor voltage}}$$

The aggressor signals are each of the DB(0-15), P_CRCA, DB(P1), and REQ or ACK pairs. If REQ is the victim line DB(0-15), P_CRCA, DB(P1), and ACK shall constitute the set of aggressor signals. If ACK is the victim line DB(0-15), P_CRCA, DB(P1), and REQ constitute the set of aggressor signals. Each aggressor signal is separately excited, the induced absolute peak noise (deviation from zero differential) on the victim pair as measured in Annex subclause E.8.

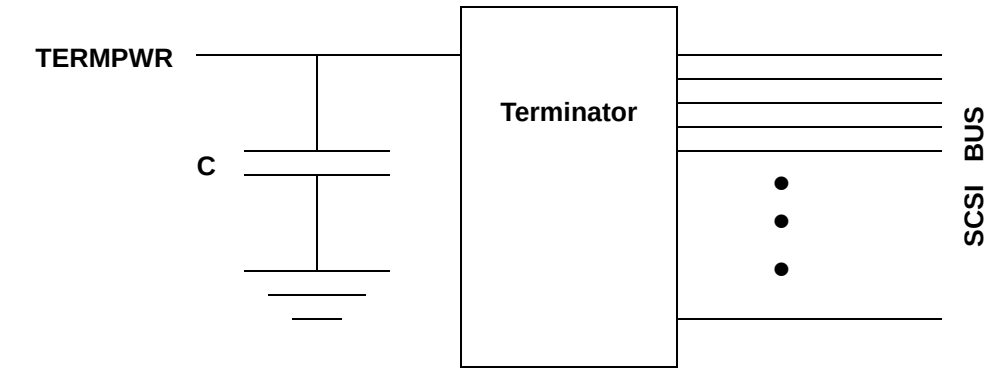
NOTE 6 - 3.0 % NEXT yields 58,9 mV peak max at 1 963 mV peak-to-peak aggressor signal amplitude

(135 Ohm maximum cable impedance at 7,3 mA max driver current). The crosstalk requirement is based only on percentage as that is all the cable influences.

6.4 Decoupling characteristics for TERMPWR lines

The TERMPWR lines should be decoupled at each terminator with at least a 2,2 μF and not greater than a 10 μF bypass capacitor.

See 7.5 and Annex D for additional information.



Note: 2,2 μF ≤ C ≤ 10 μF

Figure 26 - Terminator decoupling example

6.5 Connection requirements for RESERVED lines

The RESERVED lines shall be left open in the bus terminator assemblies and in the SCSI devices. The RESERVED lines shall have continuity from one end of the SCSI bus to the other end.

6.6 Cables used with SE transceivers

The maximum distance between terminators when using SE transceivers shall be as defined in table 12.

Table 12 - SE maximum bus path length between terminators

Number of attached SCSI devices	Maximum bus path length between terminators (meters) (note 1)		
	Fast-5	Fast-10	Fast-20 (note 2)
1 to 4 maximum capacitance SCSI devices	6	3	3
5 to 8 maximum capacitance SCSI devices	6	3	1.5
9 to 16 maximum capacitance SCSI devices	6	3	N/A

Note:
1 For environments where all elements of the bus (cables, device interfaces, environmental noise and other values) are controlled to be better than minimally required, it may be possible to extend the path length and SCSI device count (see note 11 in 7.2.4).
2 It is recommended that the SCSI devices be uniformly spaced between terminators with the end SCSI devices located as close as possible to the terminators.

The stub length when using SE transceivers shall not exceed 0,1 meter. The stub length is measured from the transceiver to the connection of the SCSI bus path (see figure 4). The spacing of SCSI devices on the SCSI bus path should be at least three times the stub length to avoid stub clustering (See Annex I).

6.7 SE ground offset

The magnitude of the ground offset voltage between logic grounds on any two device connectors shall be less than 50 mV at any frequency detectable by the receiver.

6.8 Cables used with LVD transceivers

Balanced interconnect media (e.g., twisted-planar, discrete wire twisted pairs, matched printed circuit board traces) should be used with LVD transceivers.

NOTE 7 - Use of unbalanced media such as planar untwisted construction typically produces higher crosstalk than balanced constructions but may be used if all electrical requirements are met.

The maximum distance between terminators when using LVD transceivers shall be as defined in table 13.

Table 13 - LVD maximum bus path length between terminators

Interconnect	Maximum bus path length between terminators (meters)					
	Fast-5	Fast-10	Fast-20	Fast-40	Fast-80	Fast-160
Point-to-point interconnect	25	25	25	25	25	25
Multidrop interconnect	12	12	12	12	12	12

6.9 LVD stub length and spacing

The stub length when using LVD transceivers shall not exceed 0,1 meter. The difference in stub length shall be less than 1,27 cm for the REQ, ACK, DB(15,0), P_CRCA and DB(P1) signals. Stub length differences on the plus and minus signals of the same differential line should be minimized. The stub length is measured from the stub connection (see 4.4) to the end of the stub. The spacing of SCSI devices on the SCSI bus path shall be as indicated in table 14.

Table 14 - Minimum stub connection spacing rules for LVD SCSI devices

Transmission mode	Minimum spacing between stub connections (meters)				
	40 pF/m (note)	65 pF/m (note)	90 pF/m (note)	115 pF/m (note)	140 pF/m (note)
LVD	0,36	0,22	0,16	0,13	0,10
Note: SE media capacitance with no SCSI devices attached measured per Annex E.					

7 SCSI parallel interface electrical characteristics

7.1 SCSI parallel interface electrical characteristics overview

The SCSI parallel interface may use the following transmitter implementations:

a) SE passive negation;
b) SE active negation;
c) LVD.

LVD/MSE transmitters support SE active negation. An LVD/MSE SCSI device may be damaged if exposed to voltages above 4,1 V.

If a transceiver fully complies with the requirements of more than one of the above transmitter implementations then it may interoperate with those transceiver types.

Hot plug events may cause the SCSI bus to change transmission modes. If a mode change occurs the SCSI bus is not operational until all multi-mode SCSI devices and terminators have changed modes and all other SCSI devices have released the SCSI bus.

NOTE 8 - A hot plug event that changes the SCSI bus transmission mode from LVD to SE only works if all SE requirements (e.g., SCSI device count and bus length) are met for the new configuration.

For each transmitter implementation one or more LVD receiver and capacitance specifications may apply.

For measurements in this clause, SCSI bus termination is assumed to be external to the SCSI device. See 6.5 for the termination requirements for the RESERVED lines. SCSI devices may have provision for allowing optional internal termination provided the internal termination conforms with 7.2.1, 7.3.1, or 7.4.1 when enabled and the SCSI device, including the disabled termination, conforms with 7.2.4 or 7.3.4 when the internal termination is disabled.

In addition to the SCSI device electrical requirements defined in the remaining subclauses of this clause SCSI devices shall meet the requirements specified in table 15 and table 16 at the device connector. The requirements in table 16 shall apply to both powered and unpowered SCSI devices.

Table 15 - Absolute electrical limits at the device connector

Mode	Minimum	Maximum	Notes
SE (passive negation) input voltage	-0,5 V D.C.	5,5 V D.C.	Absolute maximum at all operating conditions, for SCSI devices meeting the passive negation requirements in table 17.
SE (active negation) input voltage	-0,5 V D.C.	4,1 V D.C.	Absolute maximum at all operating conditions, for SCSI devices meeting the active negation requirements in table 17.
LVD/MSE input voltage (D.C. V + or - signal to local ground)	-0,5 V D.C.	4,1 V D.C.	Absolute maximum at all operating conditions for all signals except DIFFSENS.
DIFFSENS for LVD/MSE input voltage	-0,5 V D.C.	4,1 V D.C.	Absolute maximum at all operating conditions for the DIFFSENS connection.
Note: A combination of LVD/MSE devices and HVD devices on the same bus could result in damage to the DIFFSENS input of the LVD/MSE device.			

Table 16 - Input current requirements at the device connector for lines not being driven by the device

Transceiver Type	Maximum	Notes
MSE	± 20 µA D.C.	Measured from + or - signal $0 < V_{IN} < 3,01$ V to local ground for each signal pin.
LVD	± 20 µA D.C.	Measured from + or - signal $0 < V_{IN} < 2,5$ V to local ground for each signal pin.

7.2 SE alternative

7.2.1 SE termination

All SCSI bus signals are common among all SCSI devices connected to the bus. All signal lines shall be terminated at both ends with a terminator that is compatible with the type of transceivers used in the SCSI devices. The termination points define the ends of the bus. These termination points may be internal to one or two SCSI devices.

NOTE 9 - If the termination is provided and enabled within a SCSI device that device should not be removed from the SCSI bus while the bus is in use.

All SE conductors not defined as RESERVED, SIGNAL RETURN, or TERMPWR shall be terminated exactly once at each end of the bus. The termination of each signal shall meet the following requirements:

a) the terminators shall be powered by the TERMPWR line and may receive additional power from other sources but shall not require such additional power for proper operation (see 7.5). Direct connection between the TERMPWR source and the individual terminators without using the

TERMPWR line is also allowed;

b) each terminator shall source current to the signal line whenever its terminal voltage is below 2,5 V D.C. and this current shall not exceed 22,4 mA for any line voltage at or above 0,5 V D.C and 25,4 mA for any line voltage between 0,5 V D.C. and 0,2 V D.C. even when all other signal lines are driven at 4.0 V D.C.;
c) the voltage on all released signal lines shall be at least 2,5 V D.C.;
d) these conditions shall be met with any conforming configuration of targets and initiators as long as at least one SCSI device is supplying TERMPWR;
e) the terminator at each end of the SCSI bus (see 7.2.4) shall add a maximum of 25 pF capacitance to each signal.
f) the terminator shall not source current to the signal line whenever its terminal voltage is above 3,24 V D.C. except terminators may source current when the voltage is above 3,24 V D.C. in applications where the bus is less than 0,3 m.

Terminators employing a resistor divider network between 5 volts and ground as the termination shall not be used.

7.2.2 SE output characteristics

SE signals shall be driven with either passive-negation or, except OR-tied signals (see 8.4), active-negation drivers. Passive-negation drivers shall only be used in SCSI devices that support transfer rates no greater than fast-10. Passive-negation drivers have two states, asserted and high-impedance. Passive-negation drivers are usually implemented using an open-collector or an open-drain circuit. Active-negation drivers have three states: asserted, negated, and high-impedance. Each signal sourced by a SCSI device shall have the steady state D.C. output characteristics defined in table 17 measured at the SCSI device's connector.

Table 17 - SE steady state output voltage characteristics

Driver Type	Maximum transfer rate	SE steady state output voltage characteristics
Passive negation	**Async., Fast-5, Fast-10**	a) V_{OL} (low-level output voltage) = 0,0 V D.C. to 0,5 V D.C. at I_{OL} = 48 mA (signal asserted); b) V_{OH} (high-level output voltage) = 2,5 V D.C. to 5,25 V D.C. (signal negated)
Active negation	**Async., Fast-5, Fast-10, Fast-20**	a) V_{OL} (low-level output voltage) = 0,0 V D.C. to 0,5 V D.C. at I_{OL} = 48 mA (signal asserted); b) V_{OH} (high-level output voltage) = 2,5 V D.C. to 3,7 V D.C. (signal negated). See figure 27 for $I_{OH.}$

Passive-negation drivers do not source current to achieve the V_{OH} voltage level. They enter the high-impedance state and rely on the terminator to source the current.

The output characteristics (signal negated) for active-negation drivers shall be constrained to operate in the non-shaded areas of figure 27 for SE SCSI devices that support fast-20 and are recommended for all other SE SCSI devices.

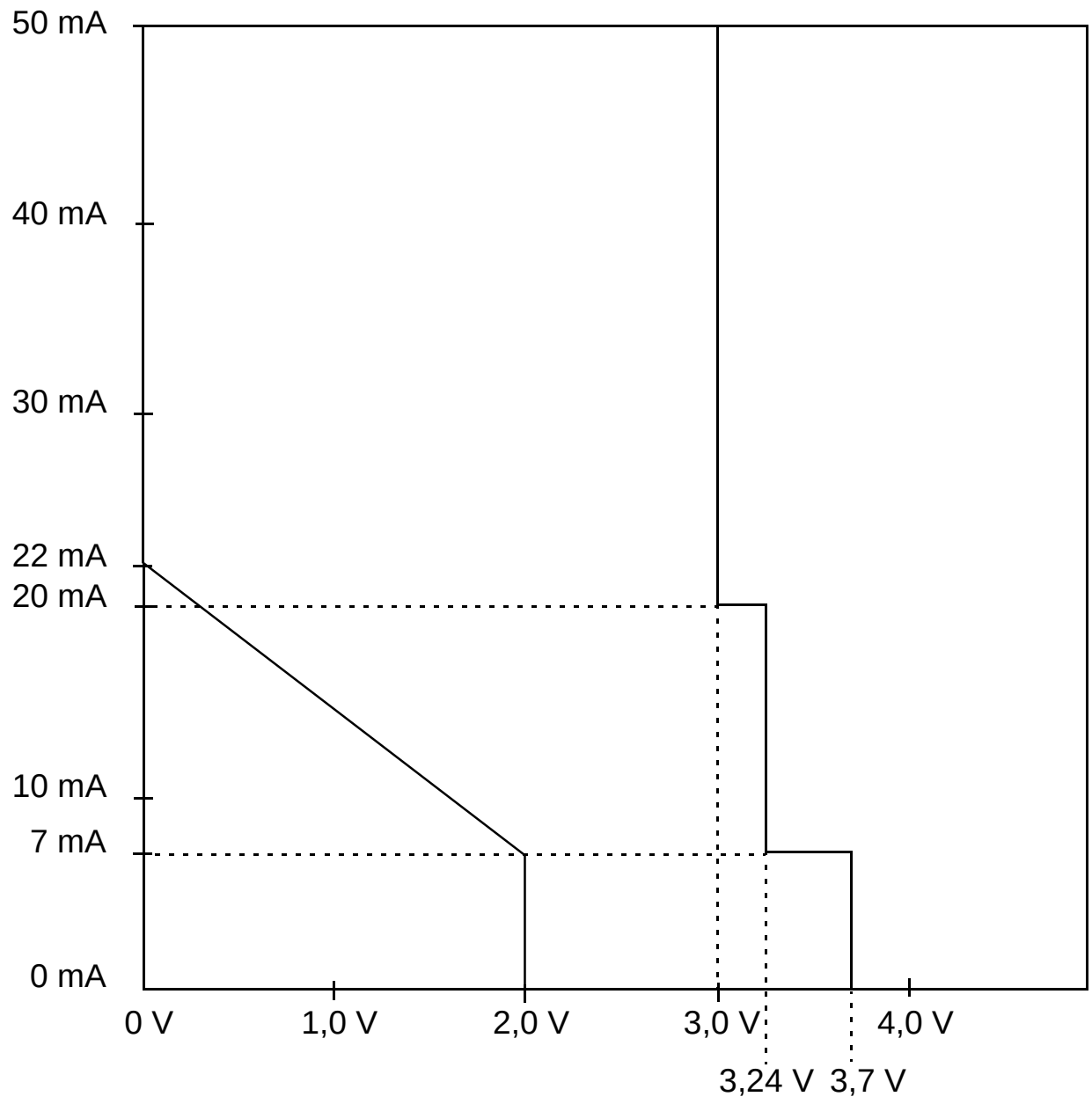


Figure 27 - Active negation current vs. voltage

NOTE 10 - Figure 27 shows the allowed domains for the D.C. output characteristics of an active-negation driver when negated. It is not intended to show A.C. output characteristics, which may be different due to other requirements such as slew rate specifications. To measure the actual SCSI device D.C. output characteristics, it is necessary to vary the SCSI device load, so the test circuit shown in figure 28 is not applicable to this measurement.

While active-negation drivers may be used on any non-OR-tied signal (see 8.4), their usage is particularly valuable on the ACK and REQ signals, because these signals are vulnerable to glitches on the transition that could lead to false ACK or REQ detection. Additional benefit may be achieved by using active-negation drivers on the DATA BUS, P_CRCA, and DB(P1) signals when operating in fast synchronous data transfer mode by reducing the skews between the first group of signals (ACK and REQ)

and the DATA BUS, P_CRCA, and DB(P1).

All SE drivers shall maintain the high-impedance state during powering on and powering off.

SCSI devices should meet the following specifications for all signals when measured on the test circuit shown in figure 28 with a load capacitor (C_L) of 15 pF ± 5 %:

a) t_{rise} (rise rate) = 520 mV per ns maximum (0,7 V D.C. to 2,3 V D.C.);
b) t_{fall} (fall rate) = 520 mV per ns maximum (2,3 V D.C. to 0,7 V D.C.).

The slew rates specified above are requirements for a driver when using the SE test circuit in figure 28. Those slew rates are not the observed rise rate or fall rate that would be observed on an actual SCSI bus.

All other A.C. output timing specifications shall be measured with the test circuit shown in figure 28 with a load capacitor (C_L) of 200 pF ± 5 %. The driver output timing using the load in the test circuit represents the timings defined in figure 46 and figure 47.

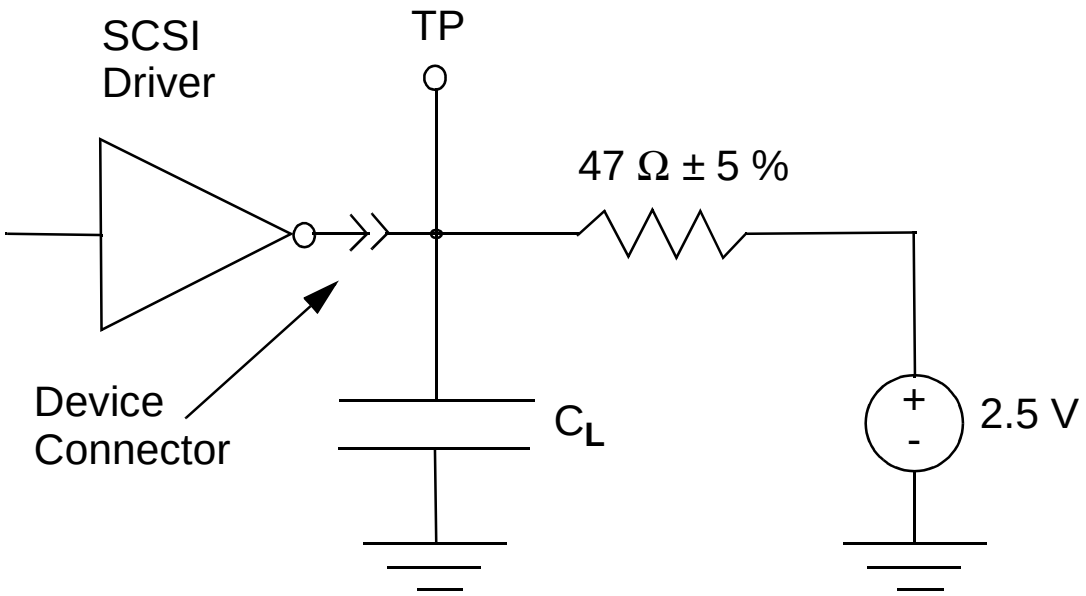


Figure 28 - SE A.C. test circuit

7.2.3 SE input characteristics

SCSI devices with power on shall meet the electrical characteristics in table 18 on each signal (including both receivers and disabled drivers). SCSI devices shall also meet the requirements in 9.3.2 and 9.3.3.

Table 18 - SE input voltage characteristics

Maximum transfer rate	SE input voltage characteristics
Fast-5	a) V_{IL} (low-level input voltage) = 0,8 V D.C. maximum (signal true); b) V_{IH} (high-level input voltage) = 2,0 V D.C.minimum (signal false) c) I_{IL} (low-level input current) = - 0,4 mA to 0,0 mA at V_I = 0,5 V D.C.; d) I_{IH} (high-level input current) = 0,0 mA to 0,1 mA at V_I = 2,7 V D.C.; e) Minimum input hysteresis = 0,2 V D.C.
Fast-10	a) V_{IL} (low-level input voltage) = 0,8 V D.C. maximum (signal true); b) V_{IH} (high-level input voltage) = 2,0 V D.C. minimum (signal false) c) I_{IL} (low-level input current) = ± 20 µA at V_I = 0,5 V D.C.; d) I_{IH} (high-level input current) = ± 20 µA at V_I = 2,7 V D.C.; e) Minimum input hysteresis = 0,3 V D.C.
Fast-20	a) V_{IL} (low-level input voltage) = 1,0 V D.C. maximum (signal true); b) V_{IH} (high-level input voltage) = 1,9 V D.C. minimum (signal false) c) I_{IL} (low-level input current) = ± 20 µA at V_I = 0,5 V D.C.; d) I_{IH} (high-level input current) = ± 20 µA at V_I = 2,7 V D.C.; e) Minimum input hysteresis = 0,3 V D.C.

Note:
1 SE input voltage characteristics specified by the maximum transfer rate supported shall apply even if a slower transfer rate is negotiated.
2 Due to the tighter voltage thresholds for fast-20, the power supply should have a maximum ±5 % tolerance of the nominal voltage.
3 All values apply to both active negation and passive negation SCSI devices.

The transient leakage current that may occur (e.g. with some ESD protection circuits) at the time of physical insertion of a SCSI device is an exponentially decaying current that does not exceed the following specifications:

a) $I_{IH.HP}$ (hot-plug high-level input current peak value excluding the first 10 ns) = +1,5 mA at V_I = 2,7 V D.C.;
b) T_{HP} (transient current duration to 10 % of peak value) = 20 µs maximum.

SCSI devices with power off should meet the above I_{IL} and I_{IH} electrical characteristics on each signal, except at time of physical insertion, when $I_{IH.HP}$ and T_{HP} prevail.

The nominal switching threshold should be 1,4 V D.C. to achieve maximum noise immunity and to ensure proper operation with complex cable configurations.

SCSI devices should incorporate a glitch filter function on REQ and ACK signals to reduce or eliminate the effect of glitch pulses.

If implemented, the glitch filter period shall not be so long as to mask out the subsequent valid transition edges of the incoming REQ and ACK signals.

7.2.4 SE input and output characteristics

The SE signals shall have the characteristics defined in table 19 when measured at the SCSI device's connector.

Table 19 - SE input and output electrical characteristics

Maximum transfer rate	SE input and output electrical characteristics
Fast-5	a) Maximum signal capacitance = 25 pF, measured at the beginning of the stub (see figure 4).
Fast-10	a) I_L (Leakage current) = -20 µA to + 20 µA at V_I = 0,0 V D.C. to 5,25 V D.C. (high-impedance state); b) Maximum signal capacitance = 25 pF, measured at the beginning of the stub (see figure 4).
Fast-20	a) I_L (Leakage current) = -20 µA to + 20 µA at V_I = 0,0 V D.C. to 4,1 V D.C. (high-impedance state); b) Maximum signal capacitance = 25 pF, measured at the beginning of the stub (see figure 4).
Note: 1 SE input and output voltage characteristics specified by the maximum transfer rate supported shall apply even if a slower transfer rate is negotiated. 2 All values apply to both active negation and passive negation SCSI devices.	

NOTE 11 - It is practical to design SCSI devices with a lumped capacitance of less than 16pF. SCSI devices without a switchable terminator may reduce this node capacitance even further. A decrease in lumped capacitance of the node and a uniform increase of the impedance along the SCSI bus towards an optimum value improves the margin and may allow for a greater number of attached SCSI devices. Backplane designs give the implementor the possibility of increasing the margins and connecting a greater number of SCSI devices to the bus.

7.3 LVD alternative

7.3.1 LVD termination

The terminators shall be powered by the TERMPWR line and may receive additional power from other sources but shall not require such additional power for proper operation.

The electrical characteristics of LVD bus termination shall be as specified in this subclause. Figure 29 shows the V_B and V_A measurement points, referenced to local ground, for the LVD bus terminator.

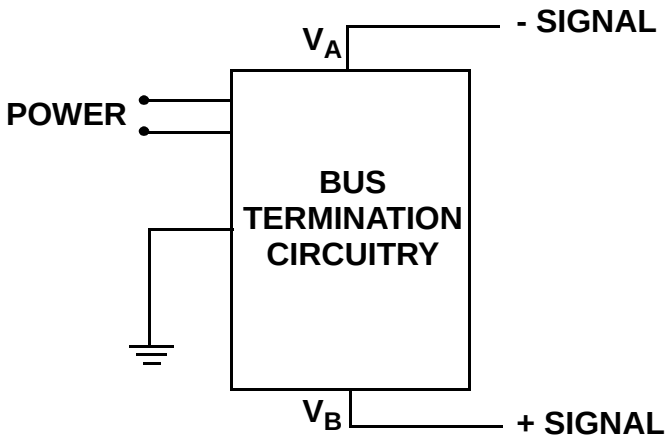


Figure 29 - LVD bus terminator

Figure 29 through figure 34 define the measurement points for the LVD terminators. Electrical characteristics shall meet the requirements in table 20 and table 21.

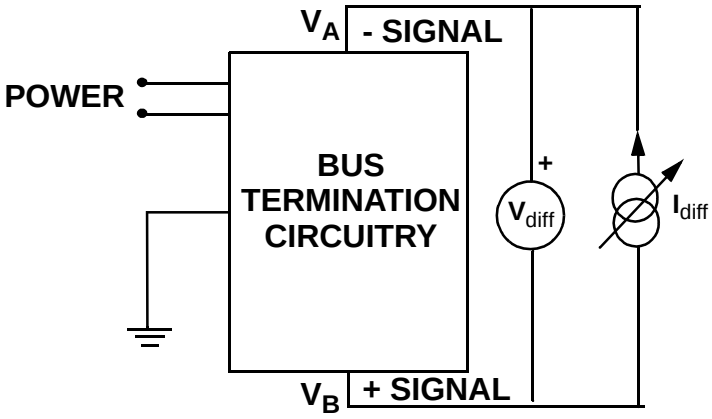


Figure 30 - Test circuit for terminator differential impedance

The requirements that relate to differential impedance are specified in figure 31 and table 20. Table 20 specifies the allowed ranges for I_{diff} and V_{diff} in figure 30. The terminator bias voltage V_{BIAS} (V_{BIAS} is the voltage measured when I = 0 in figure 31) shall have the values measured between V_1 and V_2 as measured at V_{diff} in figure 30 with the range values defined in table 20 in the LVD impedance and V_{BIAS} tests column.

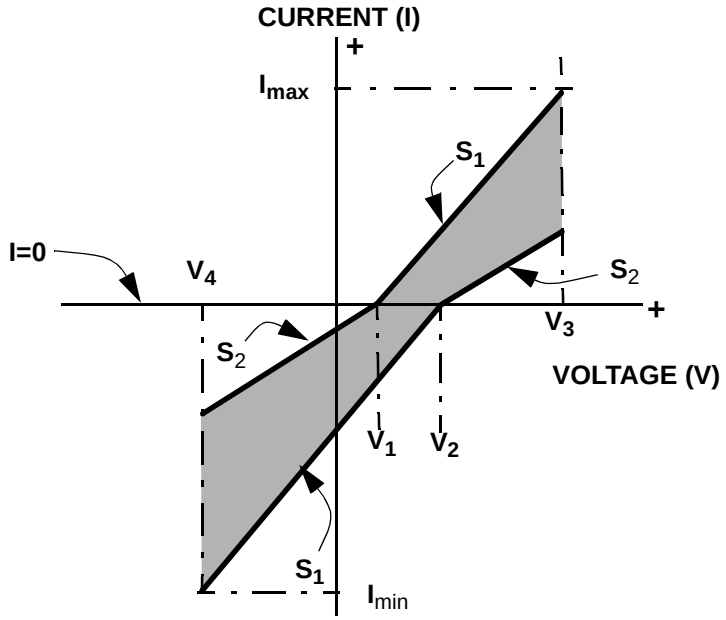


Figure 31 - Termination I-V characteristics for differential and common mode impedance tests

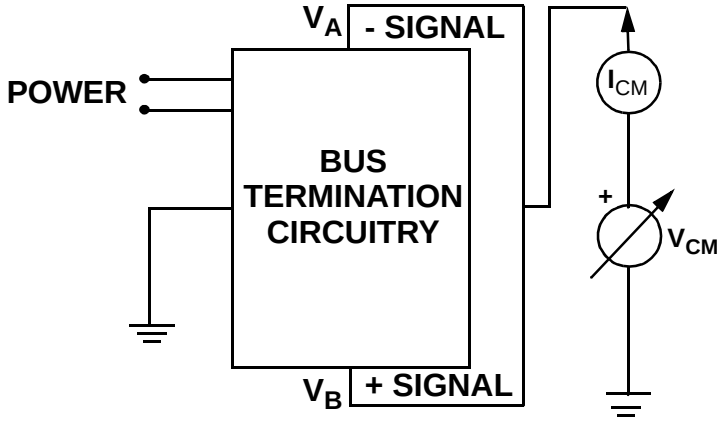


Figure 32 - Test circuit for termination common mode impedance test

The requirements that relate to common mode impedance are specified in figure 31 and table 20. Table 20 specifies the allowed ranges for I_{CM} and V_{CM} in figure 32. The terminator bias voltage V_{BIAS} (V_{BIAS} is the voltage measured when I = 0 in figure 31) shall have the values measured between V_1 and V_2 as measured at V_{CM} in figure 32 with the range values defined in table 20 in the common mode impedance and V_{BIAS} tests column.

Table 20 - I-V requirements for differential impedance, common mode impedance, and V_{BIAS} tests

Values (figure 31)	Differential impedance and V_{BIAS} tests (note) (figure 30)	Common mode impedance and V_{BIAS} tests (figure 32)
V_1 (mV)	100	1125
V_2 (mV)	125	1375
V_3 (V)	1,0	2,0
V_4 (V)	-1,0	0,5
I_{max} (mA)	9,00	N/A
I_{min} (mA)	-11,25	N/A
S_1 (Ohms)	100	100
S_2 (Ohms)	110	300
Measurement	D.C.	D.C.
Note: $V_A + V_B = 2,5 \pm 0,2$ V (figure 30)		

The requirements on termination that relate to electrical balance are specified in figure 33, figure 34, and table 21. The voltage V_{test1} in figure 33 is varied over frequencies of 0 to 40 MHz with amplitude varied over the range V_{MIN} to V_{MAX} specified in table 21 while the voltage named ΔV in figure 33 is measured. The maximum difference between values of ΔV (see figure 33) measured during this test shall be as specified in table 21.

NOTE 12 - The +SIGNAL line and -SIGNAL line capacitance should be balanced on disabled terminators.

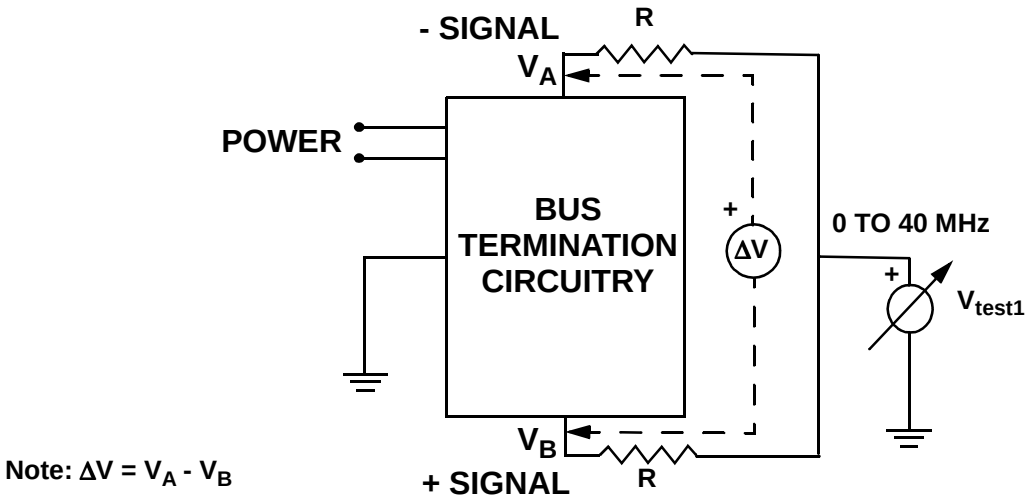


Figure 33 - Termination balance test configuration

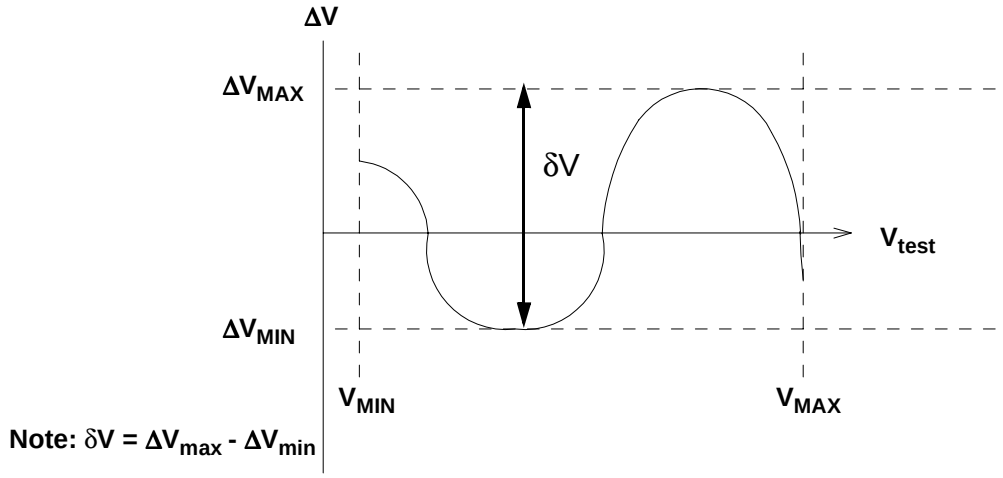


Figure 34 - Termination balance test data definition

Table 21 - Values for LVD termination balance test

Parameter	Value
V_{MIN} (V_{test1} peak)	0,7
V_{MAX} (V_{test1} peak)	1,8
R (Ohms)	100 ± 0,01%
δV	20 mV max
Note: ΔV - Input impedance for instrumentation > 10 KOhms V_{test1} swept through all values between V_{MIN} and V_{MAX}	

7.3.2 LVD driver characteristics

The LVD driver shall provide balanced asymmetrical sources that provide current from positive supply voltage to one signal line while sinking the same current to ground from the other signal line as shown in figure 36. Diagonally opposite sources operate together to produce a signal assertion or a signal negation. An assertion is produced when positive supply voltage current is sourced from SOURCE 4 to the +SIGNAL line and SOURCE 2 sinks the same current from the -SIGNAL line to ground. A negation is produced when positive supply voltage current is sourced from SOURCE 1 to the -SIGNAL line and SOURCE 3 sinks the same current from the +SIGNAL line to ground.

During paced transfers with precompensation enabled, the driver source current shall either be strong or weak to comply with precompensation requirements in table 22. The strong driver current is sourced to or from the SCSI data bus signal for the first data transfer time after a change in the data bus line state from asserted to negated or from negated to asserted. The weak driver current is sourced to or from the SCSI data bus signal after a data transfer time from the last data bus signal transition (see A.2.1). The rules for precompensation shall apply to the REQ, ACK, P1, P_CRCA and DB(15-0) signals. See figure 35 for an example of signals with and without precompensation.

Table 22 - Precompensation

	Bit pattern and driver strength							
First data bits	1	0	1	0	1	0	1	0
Driver strength	Weak	Strong	Strong	Strong	Strong	Strong	Strong	Strong
First data bits	1	1	0	1	1	0	1	1
Driver strength	Weak	Weak	Strong	Strong	Weak	Strong	Strong	Weak
First data bits	1	1	1	0	0	0	1	1
Driver strength	Weak	Weak	Weak	Strong	Weak	Weak	Strong	Weak
First data bits	0	1	0	1	0	1	0	1
Driver strength	Strong	Strong	Strong	Strong	Strong	Strong	Strong	Strong
First data bits	0	0	1	0	0	1	0	0
Driver strength	Strong	Weak	Strong	Strong	Weak	Strong	Strong	Weak
First data bits	0	0	0	1	1	1	0	0
Driver strength	Strong	Weak	Weak	Strong	Weak	Weak	Strong	Weak

Notes:
1 - In all the above examples the bit before the start of the bit pattern is a 1b
2 - The weak driver is used anytime a bit value does not change.
3 - The level of a SCSI data bus signal, after training shall follow the above rules regardless of whether or not the level is qualified as valid or invalid by the P1 signal (see 10.8.4.3). ~~The training pattern shall also follow~~ See 10.8.4.2 to determine when these precompensation ~~rules~~rules apply to the training pattern.

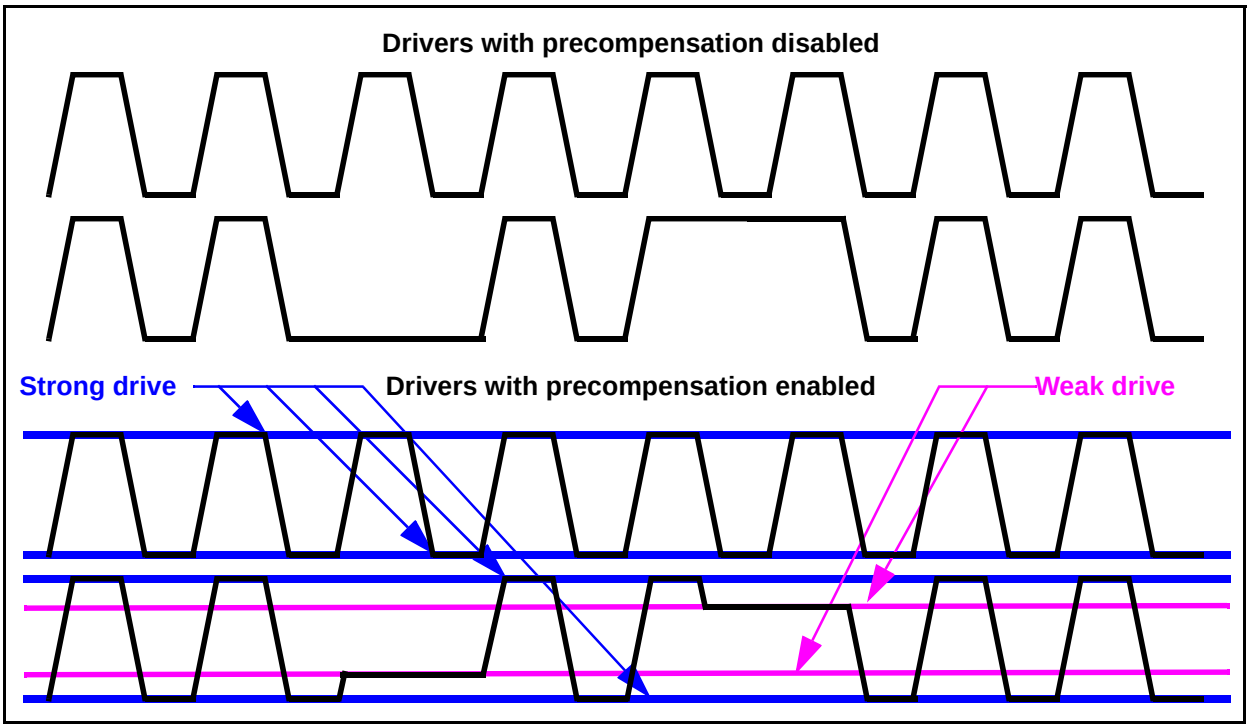


Figure 35 - Examples of driver precompensation

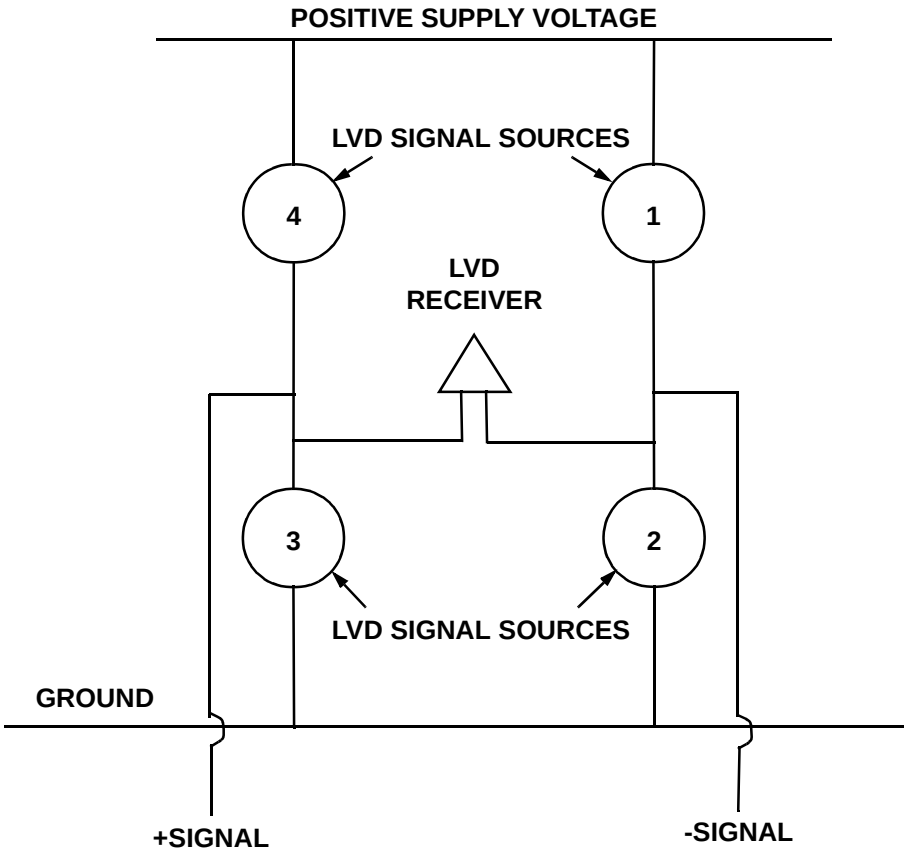


Figure 36 - LVD transceiver architecture

Balanced transmissions occur when the changes in +SIGNAL current and the changes in the -SIGNAL current precisely cancel each other. The balance is important to reduce EMI and common mode signals. Asymmetry occurs when the intensity of the SOURCE 2 and 4 assertion pair is different from the SOURCE 1 and 3 negation pair. To compensate for the negation biasing effect of the terminators, the 2 and 4 assertion pair is stronger than the 1 and 3 negation pair.

LVD drivers shall meet the requirements in Annex A.

7.3.3 LVD receiver characteristics

LVD receivers shall be connected to the +SIGNAL and -SIGNAL as shown in figure 36. An example of an LVD receiver is shown in figure 37. LVD receivers shall meet the requirements in Annex A.

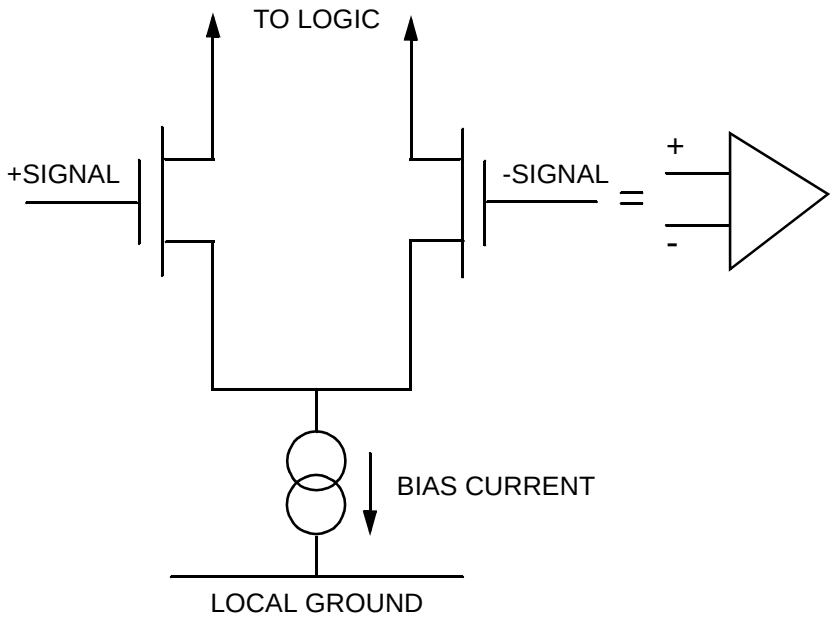


Figure 37 - LVD receiver example

7.3.4 LVD capacitive loads

Capacitive loads on differential SCSI busses shall meet the requirements specified in this section.

There are three components to differential SCSI bus capacitive loading: -SIGNAL to local ground (C1), +SIGNAL to local ground (C2), and -SIGNAL to +SIGNAL (C3) as shown in figure 38. The values C1, C2, and C3 represent measurements between the indicated points and do not represent discrete capacitors. Capacitance measurements shall be made with a nominal 1 MHz source with the same nominal D.C. level on the +SIGNAL and the -SIGNAL as specified in table 23. The driving source from the instrumentation shall apply an A.C. signal level less than 100 mV rms.

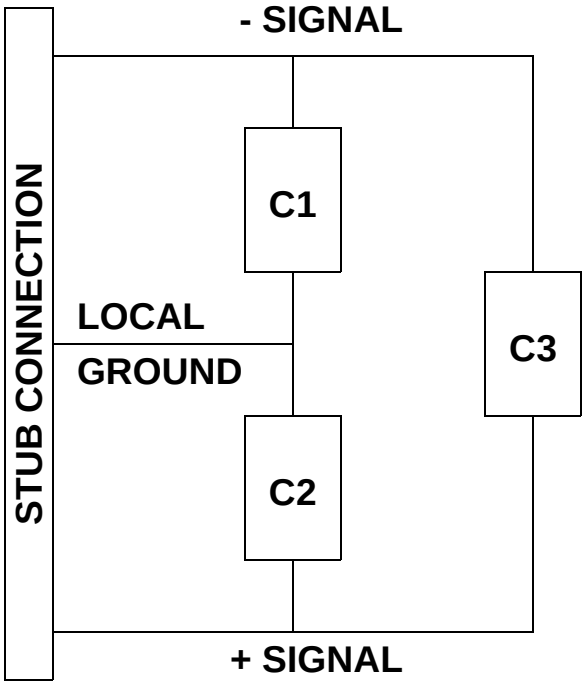


Figure 38 - LVD Capacitive loads

Table 23 - Values for LVD capacitive loads

Capacitance measurement	Maximum (note)	Description
C1 (pF)	15	@V = 0,7 to 1,8 V D.C. -sig/gnd REQ, ACK, DB(15-0), P_CRCA, and DB(P1)
C2 (pF)	15	@V = 0,7 to 1,8 V D.C. +sig/gnd REQ, ACK, DB(15-0), P_CRCA and DB(P1)
C3 (pF)	8	@V = 0,7 to 1,8 V D.C. both - and +sig/gnd V is the same for both sigs ±100mV REQ, ACK, DB(15-0), P_CRCA and DB(P1)
C1 (pF)	25	@V = 0,7 to 1,8 V D.C. -sig/gnd all other signals
C2 (pF)	25	@V = 0,7 to 1,8 V D.C. +sig/gnd all other signals
C3 (pF)	13	@V = 0,7 to 1,8 V D.C. both - and +sig/gnd V is the same for both sigs ±100mV all other signals
| C1 - C2 | (pF)	1,5	REQ, ACK, DB(15-0), P_CRCA and DB(P1) (same signal)
| C1 - C2 | (pF)	3	all other signals (same signal)
| C1(i) - C1(REQ) | (pF)	2	For DATA(i) i = 0-15 and DB(P1) and P_CRCA
| C2(i) - C2(REQ) | (pF)	2	For DATA(i) i = 0-15 and DB(P1) and P_CRCA
| C1(i) - C1(ACK) | (pF)	2	For DATA(i) i = 0-15 and DB(P1) and P_CRCA
| C2(i) - C2(ACK) | (pF)	2	For DATA(i) i = 0-15 and DB(P1) and P_CRCA

note:
It is recommended that implementors design capacitive loads to be as small as practical.

SCSI devices containing the enabled bus termination shall have maximum values 1,5 times the maximums listed in table 23. Differential bus termination circuitry that is not part of a SCSI device shall have maximum values 0,5 times the maximums listed in table 23.

7.3.4.1 Management of LVD release glitches

Under some conditions, an LVD signal that transitions from actively negated to released may cause brief pulses to the true signal state. These pulses are called “release glitches” and may last up to a bus settle delay (see 9.2.7). Requirements are defined in this subclause to avoid adverse affects from release glitches.

SCSI devices shall incorporate the requirements specified in table 24 when using LVD drivers and may incorporate the requirements when using other drivers. The usage of active negation increases crosstalk

noise margin and increases the true-to-false transition speed as compared to passive negation.

Table 24 - Glitch management requirements for SCSI devices using LVD drivers

Signals	Mode	Active negation	Transmitting device	Receiving device
BSY, SEL, RST	I,T	P	No glitch management required.	No glitch management required.
ACK, ATN during transitions to BUS FREE phase	I	R	The initiator shall wait for a BUS FREE phase (note) before releasing the ACK and ATN signals from the actively negated state.	Starting no later than a bus settle delay after releasing the BSY signal, the target shall ignore the ACK and ATN signals until a subsequent physical connect.
ACK, ATN during QAS	I	R	The initiator shall wait until two system deskew delays after it detects C/D, I/O, and MSG false before releasing the ACK and ATN signals from the actively negated state.	Starting no later than two system deskew delays after negating C/D, I/O, and MSG, the target shall ignore the ACK and ATN signals until a subsequent physical connect.
REQ during transitions to BUS FREE phase	T	R	The target shall wait 2,5 x (bus settle delay) after releasing the BSY signal before releasing the REQ signal from the actively negated state.	The initiator shall begin to ignore the REQ signal within 1,5 x (bus settle delay) of the transition of the BSY signal from true to false
REQ during QAS	T	R	The target shall wait two system deskew delays after negating C/D, I/O, and MSG before releasing the REQ signal from the actively negated state.	Starting no later than two system deskew delays (see 9.2.47) after detecting C/D, I/O, and MSG false, the initiator shall ignore the REQ signal until a subsequent physical connect.
C/D, I/O, MSG except during SELECTION and RESELECTION phases of QAS	T	R	After a SELECTION or RESELECTION phase, these signals shall not be released until the BSY signal is released.	No glitch management required.
C/D, I/O, MSG during SELECTION and RESELECTION phases of QAS	T	P	After detecting SEL true following QAS arbitration, the target shall release these signals within a QAS release delay.	No glitch management required
DATA BUS (SELECTION and RESELECTION phases)	I,T	P	The transmitting device shall release all false data bits during these phases.	No glitch management required.
DATA BUS (During information transfers)	I,T	R	No glitch management required.	No glitch management required.
Key: I = initiator; P = prohibited; R = required; T = target				
Note: BUS FREE phase starts one bus settle delay after the BSY and SEL signals are both false.				

7.3.5 SE/HVD transmission mode detection

7.3.5.1 SE/HVD transmission mode detection overview

HVD is not defined in this standard. For information on HVD SCSI device implementation see the SCSI Parallel Interface-2 standard (X3.302-1998).

Transmission mode detection by LVD SCSI devices of SE and HVD SCSI devices is accomplished through the use of the DIFFSENS line. Requirements for SCSI devices and terminators for DIFFSENS are not the same as for "signal" lines because DIFFSENS is driven and detected using its own transmission and detection scheme.

LVD termination shall drive the DIFFSENS line as specified in 7.3.5.2 and LVD SCSI devices shall sense the DIFFSENS signal as specified in 7.3.5.3.

SCSI devices and terminators connected to the DIFFSENS line shall comply with the requirements in table 15 and table 16.

7.3.5.2 LVD DIFFSENS driver

The LVD DIFFSENS driver sets a voltage level on the DIFFSENS line that uniquely defines an LVD transmission mode. LVD terminators and multimode terminators (see 7.4.1) shall provide an LVD DIFFSENS driver according to the specifications in table 25.

Table 25 - LVD DIFFSENS driver specifications

Value	max.	nominal	min.	notes
V_O (volts) when I_O = 0 to 5 mA	1,4	1,3	1,2	
I_{OS} (mA)	15	5		With TERMPWR at operational levels and V_O = 0.
| Input current D.C.| (µA)	10			With terminator disabled.
Input sink current D.C. (µA) at V_O = 2,75V	200		20	Required to prevent the line from floating and to ensure the HVD DIFFSENS drivers dominate the LVD DIFFSENS drivers.

Note:
- All requirements apply at the terminator bussing connection (see figure 4).
- All measurements per figure 39.
- I_{OS} = I_O short circuit, when SE SCSI device is attached.

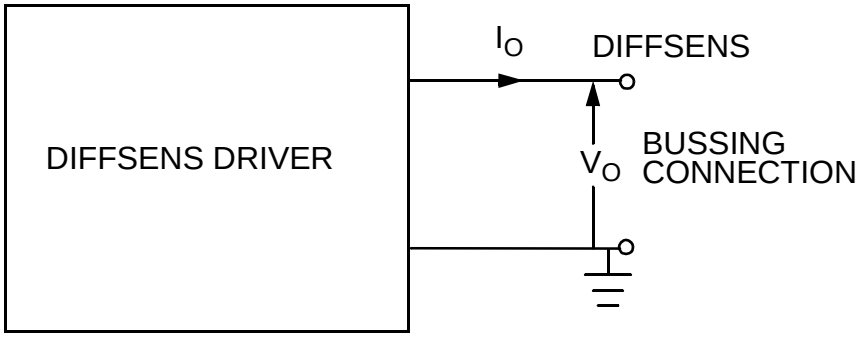


Figure 39 - LVD DIFFSENS driver signal definitions

7.3.5.3 LVD DIFFSENS receiver

LVD SCSI devices shall incorporate an LVD DIFFSENS receiver that detects the voltage level on the DIFFSENS line for purposes of informing the SCSI device of the transmission mode being used by the bus. The LVD DIFFSENS receiver shall be capable of detecting SE, LVD, and HVD SCSI devices. Table 26 and figure 40 define the receiver input levels for each of the three modes.

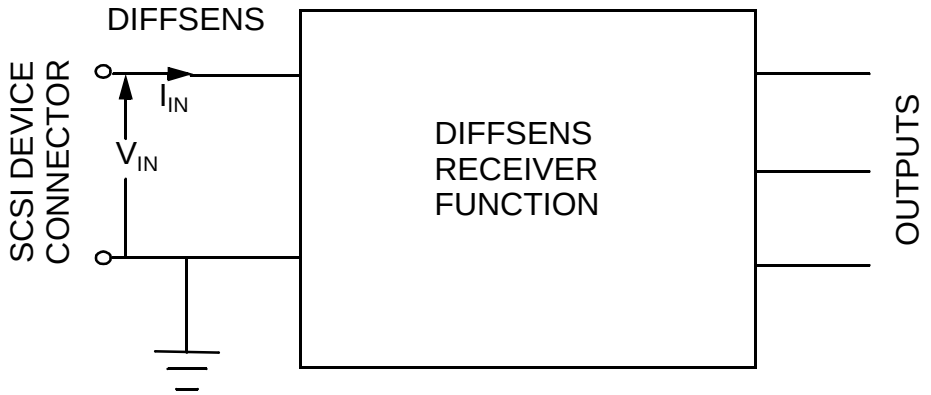


Figure 40 - DIFFSENS receiver function

Table 26 - DIFFSENS receiver operating requirements

V_{in} range	Sensed differential driver type
-0,35 V to +0,5 V	SE
> +0,5 V to < 0,7 V	indeterminate for detecting SE and LVD driver type
0,7 V to 1,9 V	LVD
> 1,9 V to < 2,4 V	indeterminate for detecting LVD and HVD driver type
2,4 V to 5,5 V	HVD

Note:
1 Input resistance (Vin/Iin) shall be 200 KOhms to 250 KOhms @ Vin < 2,7 V under all conditions of power supply (i.e., powering on, powering off, power transients)
2 All voltages measured at the device connector with respect to local ground.
3 A combination of LVD/MSE devices and HVD devices on the same bus could result in damage to the DIFFSENS input of the LVD/MSE device.

The input resistance requirement is for purposes of providing ground reference if no DIFFSENS drivers are connected to the bus and to ensure that the DIFFSENS receivers do not load the DIFFSENS drivers excessively and to ensure that SE mode is detected.

SCSI devices shall not allow the +SIGNAL line or -SIGNAL line drivers to leave the high impedance state during initial power on until both of the following conditions are satisfied:

a) the SCSI device is capable of logical operation for at least a DIFFSENS voltage filter time (see 9.2.20), and

NOTE 13 - The DIFFSENS voltage filter time delay allows time for the DIFFSENS pin to settle after the initial power connection (in the case of insertion of a SCSI device into an active system), or allows time for the power distribution system to settle.

b) the DIFFSENS mode detected has remained stable for an additional 100 ms after (a) is achieved.

A SCSI device shall not change its present signal driver or receiver mode based on the DIFFSENS voltage level unless a new mode is sensed continuously for at least a DIFFSENS voltage filter time. A multimode SCSI device shall change to the new signal driver or receiver mode based on the DIFFSENS voltage level within 400 ms of the last DIFFSENS voltage change regardless of the DIFFSENS voltage filter time. SCSI devices not capable of the new mode shall release the SCSI bus and remain in the high impedance state after the DIFFSENS voltage filter time.

An example implementation of an LVD DIFFSENS receiver is shown in figure 41. The reference voltage tolerance is greater on the higher voltage reference. This allows a simple resistor divider between V_{CC} and ground for the references. The DIFFSENS voltage filter time requirement is implemented in logic in this example and is not shown in the figure 41.

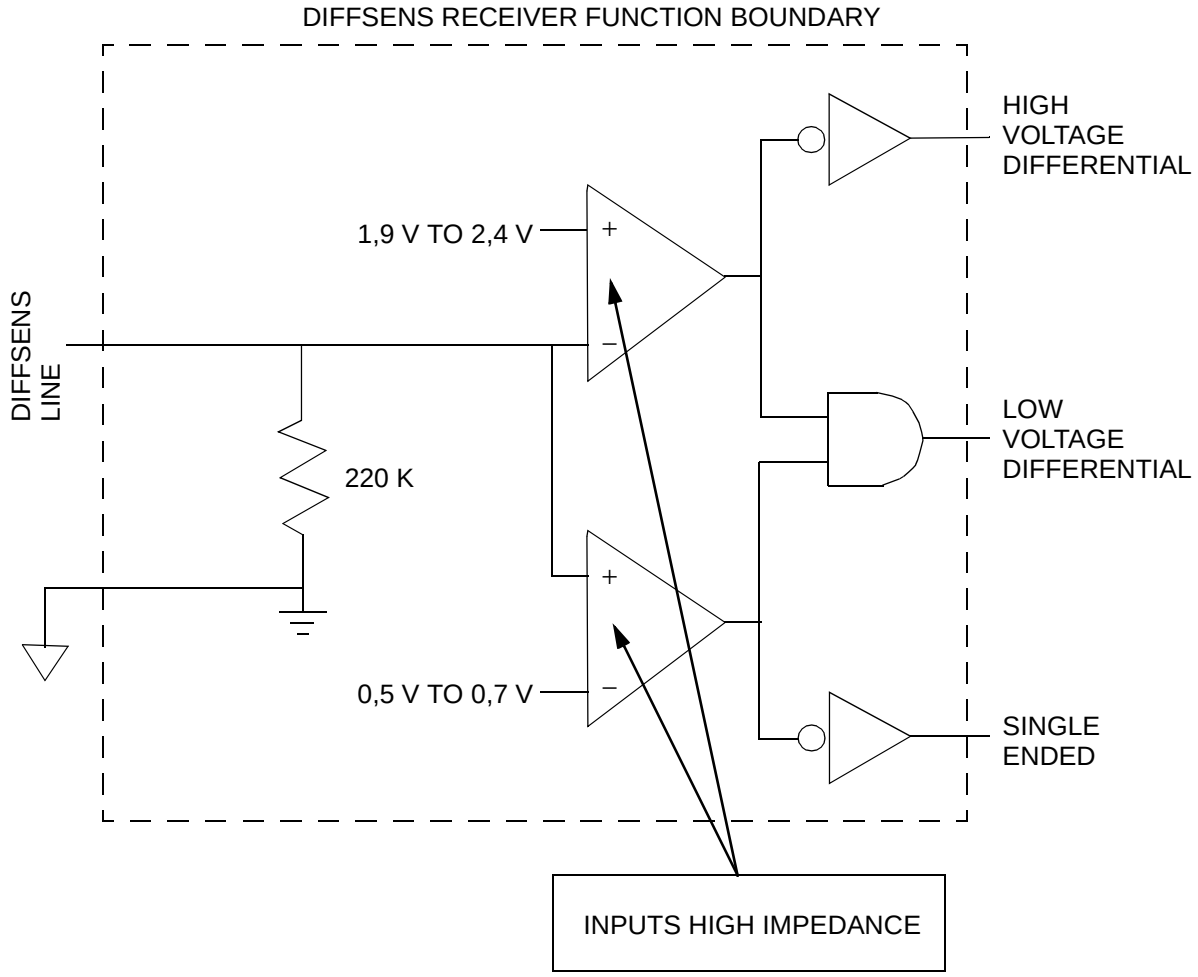


Figure 41 - LVD DIFFSENS receiver example

7.4 LVD/MSE multimode alternative

7.4.1 LVD/MSE multimode termination

Figure 42 shows the architecture of the multimode terminator.

Multimode terminators sense the DIFFSENS line while sourcing the DIFFSENS signal with the LVD levels (see table 25). The DIFFSENS line being grounded indicates that one or more SE SCSI devices or SE terminators are attached to the bus. A multimode terminator shall switch to the termination mode that is appropriate for the bus based on the value of the DIFFSENS input voltage. The appropriate mode is indicated in table 26.

A multimode terminator shall not change its present termination mode based on the DIFFSENS voltage level unless a new mode is sensed continuously for at least a DIFFSENS voltage filter time (see 9.2.20). A multimode terminator shall change to the new termination mode based on the DIFFSENS voltage level within 400 ms of the last DIFFSENS voltage change regardless of the DIFFSENS voltage filter time.

When operating in the LVD mode the requirements in 7.3 and LVD requirements in table 15 and table 16 shall apply. When operating in the MSE mode SE termination requirements in 7.2 and MSE requirements in table 15 and table 16 shall apply.

Multimode terminators are required to provide a ground driver (similar to that described for multimode transceivers) for purposes of establishing a ground reference for the SE transmission lines. The ground driver shall turn on and remain on while the DIFFSENS line indicates SE operation. When turned on ground drivers shall appear resistive with the following performance requirements:

<0,5 V @ +5 mA, > -0,5 V @ -5 mA.

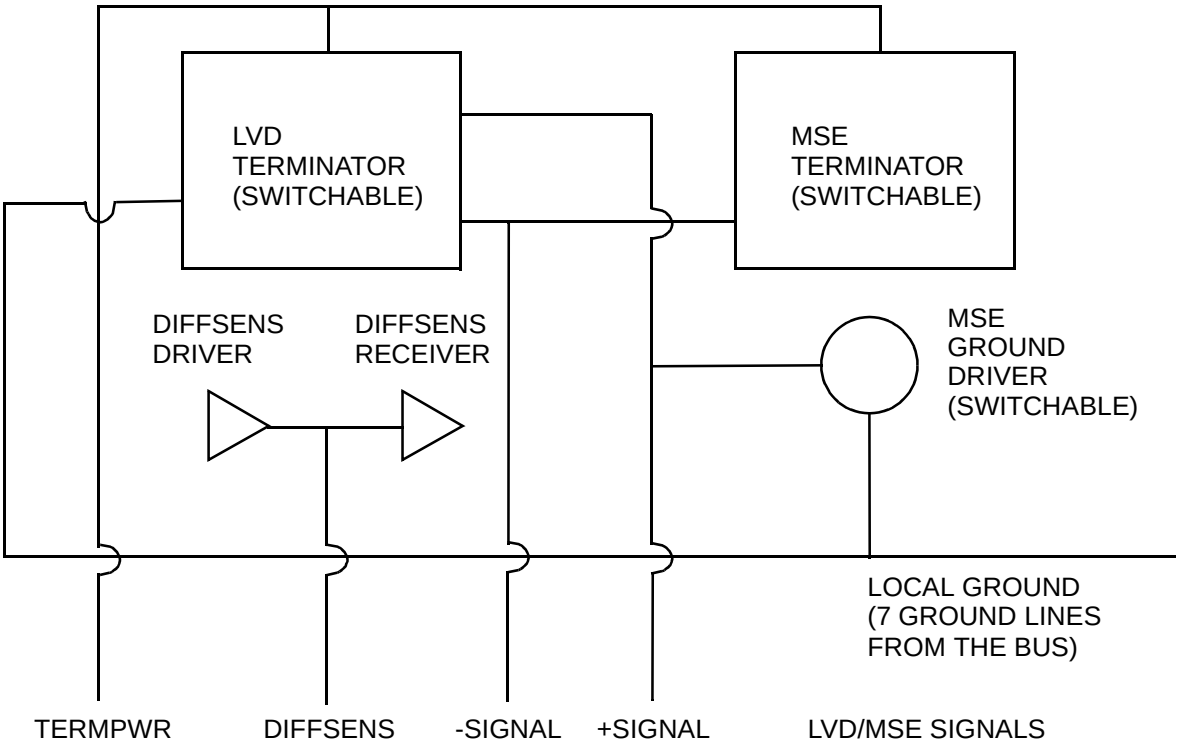
The requirements for the multimode terminator ground driver are different from those for the multimode transceiver ground driver (see 7.4.3) because the SCSI devices provide the bulk of the grounding. SCSI devices may be located far from the ends of the bus where ground references are more important.

NOTE 14 - If there is at least one SE SCSI device or terminator then there is at least one hard ground on each +SIGNAL line when operating in MSE mode (caused by the SCSI devices and/or terminators that are single-ended). This hard ground provides a return path for any low frequency currents in the +SIGNAL line.

NOTE 15 - The +SIGNAL line and -SIGNAL line capacitance should be balanced on disabled terminators.

When operating in an HVD environment the voltage on a termination contact may reach as high as 15 V above local ground due to allowed common mode transients for HVD. Multimode termination is not recommended for environments with common mode voltages exceeding the safe levels for SE and LVD SCSI devices (see table 15 and table 16).

NOTE 16 - When using only the SCA-2 connector (see 5.2.4) the SE, LVD, and HVD connector contact numbers accommodate all three modes. Switching between the LVD and the SE modes may occur with multi-mode devices if so indicated by the DIFFSENS line. HVD mode requires all devices to be HVD devices.



LOGIC CONNECTIONS NOT SHOWN

Figure 42 - Multimode terminator architecture

7.4.2 LVD/MSE multimode transceiver characteristics

The architecture for the multimode transceiver is shown in figure 43.

The contact assignments in table 5, table 6, table 7, and table 8 provide compatible alignment of the pins between SE and LVD for all connector alternates. This alignment allows a single interface to supply MSE and LVD transceivers within the same SCSI device.

When operating in an HVD environment the voltage on a transceiver contact may reach as high as 15 V above local ground due to high allowed common mode transients for HVD. LVD/MSE multimode transceivers are not recommended for exposure to environments with common mode voltages exceeding the safe levels unless the common mode voltages in the environment are controlled to safe levels for SE, LVD, and LVD/SE multimode SCSI devices (see table 15 and table 16).

An LVD/MSE multimode SCSI device shall meet the timing requirements of the DIFFSENS receiver in 7.3.5.3.

LVD/MSE multimode transceivers shall be set to the appropriate mode by sensing the output of the DIFFSENS receiver. If the DIFFSENS receiver indicates SE the LVD/MSE multimode transceiver shall follow the SE requirements in 7.2 and MSE requirements in table 15 and table 16. If the DIFFSENS line indicates LVD mode the LVD/MSE multimode transceiver shall follow the requirements in 7.3. If HVD operation is indicated by the DIFFSENS receiver all signals (except DIFFSENS) shall be set to a high impedance state (> 100 K Ohms to the local ground).

NOTE 17 - Protocol chips may offer SE signals that exceed fast-20 data rates (e.g., fast-40) to drive separate LVD or HVD transceivers. These signals are not specified for direct connection to a SCSI bus.

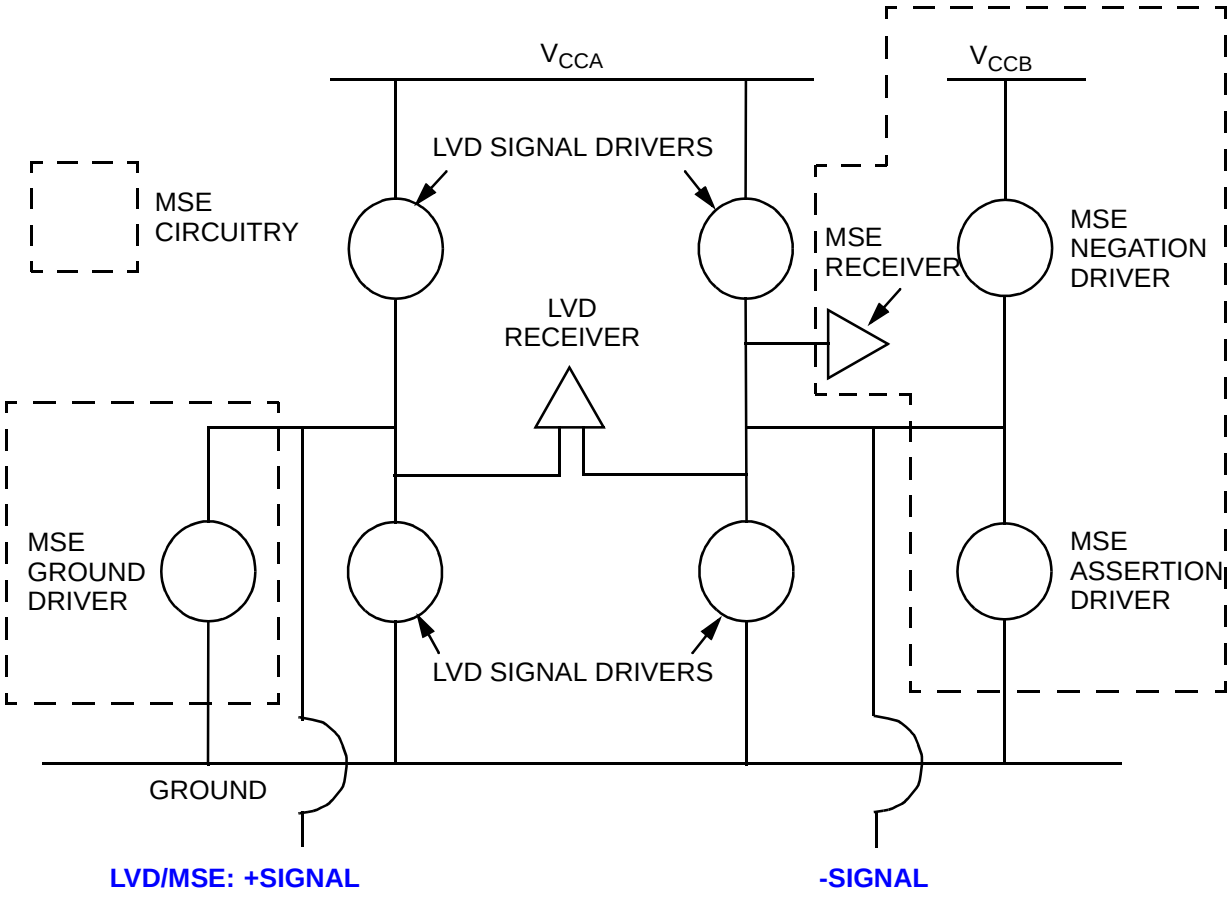


Figure 43 - Multimode transceiver architecture

7.4.3 Transceiver ground drivers

When using the multimode driver architecture described in figure 43 an MSE driver is required for the ground side of the driver. The ground driver provides the connection to ground for the SE ground line associated with the SE -SIGNAL line. When the MSE alternate is implemented the signal ground connections should be through the ground drivers to allow switching to LVD mode.

When turned on, ground drivers shall appear resistive with the following performance requirements: < 0,5 V @ +25 mA, > -0.5 V @ -25 mA. Ground drivers shall remain on for the entire time the SCSI device is powered on and is sensing a SE transmission mode from the DIFFSENS receiver as specified in 7.3.5.3. ground drivers are not required to implement any slew rate controls.

NOTE 18 - The disabled ground driver capacitance is a factor in meeting the capacitance requirements of table 23.

7.5 Terminator power

Provision shall be made to provide power from one or more sources to the TERMPWR lines of the SCSI bus. Terminator power shall be supplied through a low forward drop diode or similar semiconductor that prevents backflow of power if one of the sources of TERMPWR is powered off.

Bus terminators shall be powered from at least one source of termination power (TERMPWR). The TERMPWR lines in the cable are available for distribution of termination power. Direct connection between the TERMPWR source and the individual terminators without using the TERMPWR line is also allowed.

If the TERMPWR source is connected to the cable TERMPWR line, the source shall be isolated in a manner that prevents sinking of current from the TERMPWR line into the TERMPWR source (for example if the TERMPWR source voltage falls below the voltage existing on the TERMPWR line, the TERMPWR source sinks current unless unidirectional isolation is present in the TERMPWR source).

Regulatory agencies may require limiting maximum (short circuit) current to the TERMPWR lines. These requirements generally mandate the use of current limiting circuits and may restrict the number of sources provided for TERMPWR.

The terminator power characteristics, for each terminator, as measured at the terminator bus path connector shall be as defined in table 27.

Table 27 - Terminator power characteristics at the terminator

Terminator power characteristics	Terminator type				
	SE			LVD	SE/LVD (Multimode)
	A cable	P cable			
		0,2 V dropout regulator			
I_{min} (A) @ V_{min}	0,35	0,6	0,6	0,5	0,65
V_{min} (V) @ I_{min}	4,0	2,7	4,0	3,0	3,0
V_{max} (V) @ all conditions	5,25	5,25	5,25	5,25	5,25
Note: The recommended TERMPWR current limiting is 2,0 amps.					

NOTE 19 - SCSI devices connected with a maximum length SE A cable (table 3) are not able to meet the source current requirements in table 27 unless the TERMPWR conductor size is 0,080 98 mm^2 (28 AWG) minimum because the SE A cable contains only one TERMPWR line.

NOTE 20 - It is recommended that a SCSI device connected with the nonshielded alternative 2 connectors (see 5.2.2) that provide terminator power use keyed connectors to prevent accidental grounding or the incorrect connection of terminator power.

It is recommended that the terminator power lines be decoupled at each terminator with a bypass capacitor of at least 2,2 µF, to improve signal quality, but not greater than 10 µF. (see 6.4)

Usage of the terminator power lines for distribution of power for purposes other than for SCSI bus termination is outside the scope of this standard.

8 SCSI bus signals

8.1 SCSI bus signals overview

Information transfer on the SCSI bus is allowed between only two SCSI devices at any given time except during MESSAGE IN phase when QAS is enabled (see 16.3.10). All SCSI devices that have QAS enabled are required to monitor messages during a MESSAGE IN phase for a QAS REQUEST message. The maximum number of SCSI devices is determined by the width of the data path implemented and restrictions in clause 7. The SCSI devices may be any combination of initiators and targets, provided there is at least one of each.

Each SCSI device has a SCSI address and a corresponding SCSI ID bit assigned to it. When two SCSI devices communicate on the SCSI bus, one acts as an initiator and the other acts as a target. An initiator originates an I/O process and the target performs the I/O process.

NOTE 21 - A SCSI device is usually fixed as an initiator or target, but some SCSI devices may be able to assume either role.

Table 28 shows the relationship between SCSI Addresses, SCSI IDs, and arbitration priority. In table 28 a hyphen (“-”) represents a logical zero bit resulting from the data bus bit being released.

.

Table 28 - Arbitration priorities by SCSI ID

SCSI address	DB 15							DB 8	DB 7							DB 0	Priority
7	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	1
6	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	2
5	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	3
4	-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	4
3	-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	5
2	-	-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	6
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	-	7
0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	8
15	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	9
14	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	10
13	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	11
12	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	12
11	-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	13
10	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	14
9	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-	15
8	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	16

8.2 Signal descriptions

BSY (BUSY). An "OR-tied" signal that indicates that the SCSI bus is in use.

SEL (SELECT). An "OR-tied" signal used by an initiator to select a target or by a target to reselect an initiator.

RST (RESET). An "OR-tied" signal that indicates the RESET condition.

C/D (CONTROL/DATA). A signal sourced by a target that indicates whether control or DATA phase information is on the DATA BUS. Asserted indicates control (i.e., COMMAND, STATUS, and MESSAGE phases).

I/O (INPUT/OUTPUT). A signal sourced by a target that controls the direction of data movement on the DATA BUS with respect to an initiator. Asserted indicates INPUT. This signal is also used to distinguish between SELECTION and RESELECTION phases.

MSG (MESSAGE). A signal sourced by a target to indicate the MESSAGE phase or a DT DATA phase depending on whether C/D is true or false. Asserted indicates MESSAGE or DT DATA.

REQ (REQUEST). A signal sourced by a target to indicate a request for an information transfer on the SCSI bus.

ACK (ACKNOWLEDGE). A signal sourced by an initiator to respond with an acknowledgment of an information transfer on the SCSI bus.

ATN (ATTENTION). A signal sourced by an initiator to indicate the ATTENTION condition.

P_CRCA (PARITY/CRC AVAILABLE) . A signal indicating either parity or CRC available based on bus phase and negotiated settings.

When referred to as DB(P_CRCA) it is a signal sourced by the SCSI device driving the DATA BUS during the SELECTION phase, ST DATA phase, COMMAND phase, MESSAGE phase, and STATUS phase. The P_CRCA signal is associated with the DB(7-0) signals and is used to detect the presence of an odd number of bit errors within the byte. The P_CRCA bit is driven such that the number of logical ones in the byte plus the parity bit is odd.

When referred to as P_CRCA and data group transfers are enabled (see 16.3.10) it is a signal sourced by a target during DT DATA phases to control whether a data group field is a pad field, pCRC field, or data field (see 10.8.3.3.5). When asserted the data group field shall be pad or pCRC fields that shall not be transferred to the application client. When negated the data group field shall be a data field that shall be transferred to the application client.

When referred to as P_CRCA and information unit transfers are enabled with write flow control and read streaming disabled it is a signal sourced by a target that shall be continuously negated by the target and shall be ignored by the initiator during DT DATA phases.

~~NOTE 22 - The P_CRCA signal in previous versions of this standard was labeled as the DB(P0) signal.~~

When referred to as P_CRCA and information unit transfers are enabled with write flow control enabled it is a signal sourced by a target that shall be continuously negated by the target and shall be ignored by the initiator during DT DATA OUT phases. During DT DATA IN phases it is a signal sourced by a target. An asserted P_CRCA indicates the current SPI data stream information unit is the last SPI data stream information unit of the current write stream.

When referred to as P_CRCA and information unit transfers are enabled with read streaming enabled it is a signal sourced by a target that shall be continuously negated by the target and shall be ignored by the initiator during DT DATA IN phases. During DT DATA OUT phases it is a signal sourced by a target. An asserted P_CRCA indicates the current SPI data stream information unit is the last SPI data stream information unit of the current read stream.

P1 (PARITY 1). A signal normally sourced by the SCSI device driving the DATA BUS. The P1 signal is associated with the DB(15-8) signals and is used to detect the presence of an odd number of bit errors within the byte. The P1 bit is driven such that the number of logical ones in the byte plus the P1 bit is odd.

During the SELECTION phase, ST DATA phase with transfer length set for 8-bit transfers, COMMAND

phase, MESSAGE phase, and STATUS phase the P1 signal shall not be driven by any SCSI device.

When data group transfers are enabled (see 16.3.10) the P1 signal shall be continuously negated by the SCSI device driving the DB(15-0) signals and shall be ignored by the SCSI device receiving the DB(15-0) signals during DT DATA phases.

When information unit transfers and synchronous transfers are enabled the P1 signal shall be continuously negated by the SCSI device driving the DB(15-0) signals and shall be ignored by the SCSI device receiving the DB(15-0) signals during DT DATA phases.

When information unit transfers and paced transfers are enabled the P1 signal shall be sourced by the SCSI device driving the DATA BUS. The P1 signal is used to indicate whether the data valid or data invalid state during paced transfers (see 10.8.4.3).

DB(7-0) (8-bit DATA BUS). Eight data-bit signals that form the 8-bit DATA BUS. Bit significance and priority during arbitration are shown in table 28.

DB(15-0) (16-bit DATA BUS). Sixteen data-bit signals that form the 16-bit DATA BUS. Bit significance and priority during arbitration are shown in table 28.

8.3 Signal states

8.3.1 SE signals

Signals may be in a true (asserted) or false (negated) state. Signals that are asserted are actively driven to the true state. Signals that are negated may either be actively driven to the false state or released to the false state. A signal that is released goes to the false state because the bias of the terminator pulls the signal false. OR-tied signals shall not be actively driven false.

NOTE 23 - The advantage of actively negating signals false during information transfer is that the noise margin is higher than if the signal is simply released. This facilitates reliable data transfer at high transfer rates.

Bits of the DATA BUS are defined as one when the signal is true, and defined as zero when the signal is false.

8.3.2 LVD signals

Figure 44 defines the voltage and current definitions. A signal that is released goes to the false state because the bias of the terminator pulls the signal false.

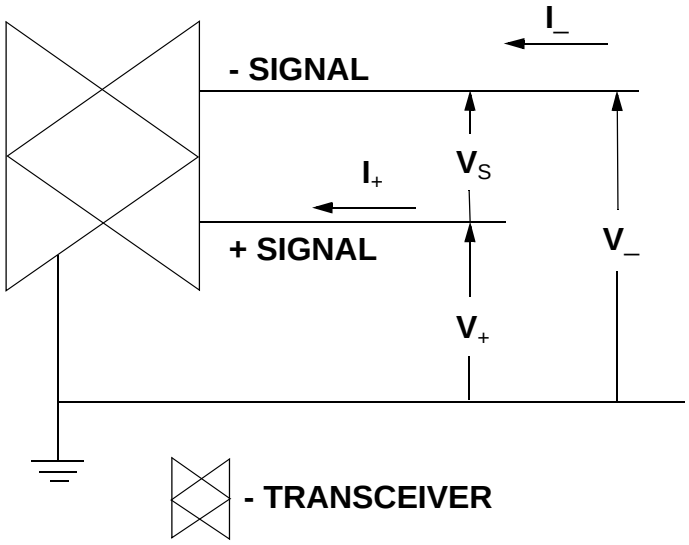


Figure 44 - Voltage and current definitions

Figure 45 defines the signaling sense of the voltages appearing on the - signal and + signal lines as follows:

a) The - signal terminal of the driver shall be negative with respect to the + signal terminal for an asserted state.
b) The - signal terminal of the driver shall be positive with respect to the + signal terminal for a negated state.

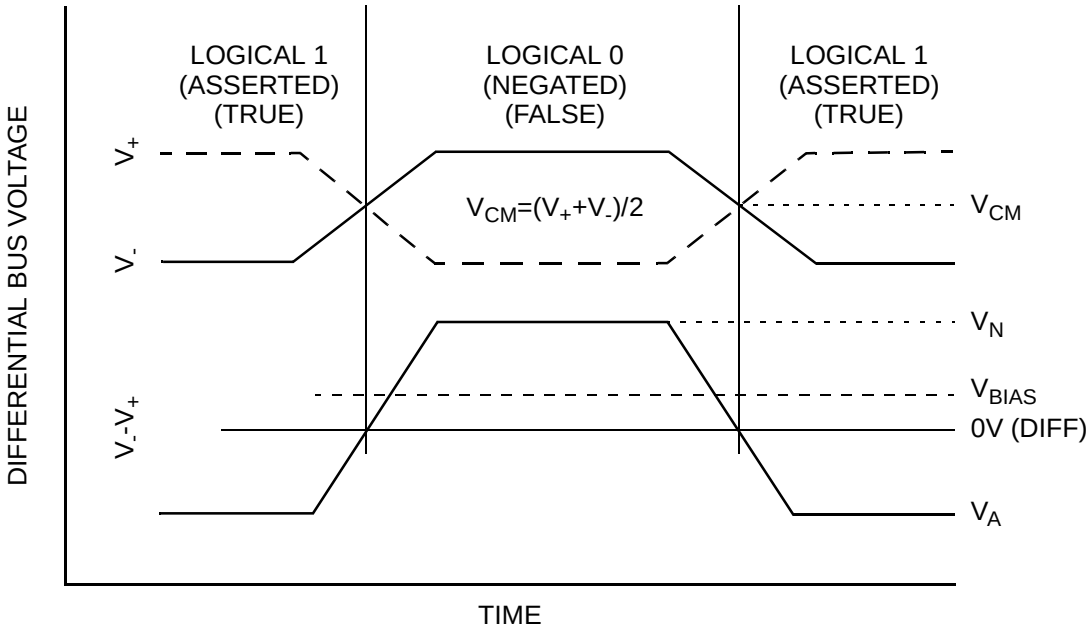


Figure 45 - LVD Signaling sense

NOTE 24 - For a description of V_{BIAS} see 7.3.1.

8.4 OR-tied signals

The BSY, SEL, and RST signals shall be OR-tied.

BSY and RST signals may be simultaneously driven true by several SCSI devices. No signals other than BSY, SEL, RST, DB(P_CRCA), and DB(P1) are simultaneously driven by two or more SCSI devices. DB(P_CRCA), and DB(P1) shall not be driven false during the ARBITRATION phase but may be driven false in other phases.

8.5 Signal sources

Table 29 indicates the type of SCSI device allowed to source each signal. No attempt is made to show if the source is driving asserted, driving negated, or is released. All SCSI device drivers that are not active sources shall be in the high-impedance state. The RST signal may be asserted by any SCSI device at any time.

Table 29 - Signal sources

SCSI bus phase	P cable signals						
	A cable signals						
	BSY	SEL	C/D I/O MSG REQ	ACK ATN	DB(7-0)	P_CRCA	DB15-8 DB(P1)
BUS FREE	None	None	None	None	None	None	None
Normal ARBITRATION	All	Win	None	None	S ID	S ID	S ID
QAS ARBITRATION	PT	Win	None	None	S ID	S ID	S ID
SELECTION	I&T	Init	None	Init	Init	Init	Init
RESELECTION	I&T	Targ	Targ	Init	Targ	Targ	Targ
COMMAND	Targ	None	Targ	Init	Init	Init	None
ST DATA IN	Targ	None	Targ	Init	Targ	Targ	Targ
ST DATA OUT	Targ	None	Targ	Init	Init	Init	Init
DT DATA IN	Targ	None	Targ	Init	Targ	Targ	Targ
DT DATA OUT	Targ	None	Targ	Init	Init	Targ	Init
STATUS	Targ	None	Targ	Init	Targ	Targ	None
MESSAGE IN	Targ	None	Targ	Init	Targ	Targ	None
MESSAGE OUT	Targ	None	Targ	Init	Init	Init	None

All: The signal shall be driven by all SCSI devices that are actively arbitrating.

S ID: A unique data bit (the SCSI ID) shall be driven by each SCSI device that is actively arbitrating; the other data bits shall be released (i.e., not driven) by this SCSI device. The P_CRCA and DB(P1) bit(s) may be released or driven to the true state, but shall not be driven to the false state during this phase.

I&T: The signal shall be driven by the initiator, target, or both, as specified in the SELECTION phase and RESELECTION phase.

Init: If driven, the signal shall be driven only by the active initiator.

None: The signal shall be released; that is, not driven by any SCSI device. The bias circuitry of the bus terminators pulls the signal to the false state.

Win: The signal shall be driven by the one SCSI device that wins arbitration (see 10.5).

Targ: If driven, the signal shall be driven only by the active target.

PT: The signal shall be driven by the target that initiated the QAS arbitration (see 10.5).

9 SCSI parallel bus timing

9.1 SCSI parallel bus timing values

See table 30, table 32, and table 33 SCSI bus timing values. Unless otherwise indicated, the ~~delay time~~ timing measurements for each SCSI device, shown in table 30, shall be calculated from signal conditions existing at that SCSI device's port. The timing characteristics of each signal are described in the following paragraphs. Timing requirements relating to LVD release glitches are defined in 7.3.4.1.

Table 30 - SCSI bus control timing values

Subclause	Timing description	Type	Timing values
9.2.1	Arbitration delay	minimum	2,4 µs
9.2.4	Bus clear delay	maximum	800 ns
9.2.5	Bus free delay	minimum	800 ns
9.2.6	Bus set delay	maximum	1,6 µs
9.2.7	Bus settle delay	minimum	400 ns
9.2.8	Cable skew (note 1)	maximum	4 ns
9.2.19	Data release delay	maximum	400 ns
9.2.20	DIFFSENS voltage filter time	minimum	100 ms
9.2.22	Physical disconnection delay	minimum	200 us
9.2.23	Power on to selection (note 2)	maximum	10 s
9.2.24	QAS arbitration delay	minimum	1000 ns
9.2.25	QAS assertion delay	maximum	200 ns
9.2.26	QAS release delay	maximum	200 ns
9.2.27	QAS non-DATA phase REQ (ACK) period	minimum	50 ns
9.2.40	Reset delay	minimum	200 ns
9.2.41	Reset hold time	minimum	25 µs
9.2.42	Reset to selection (note 2)	maximum	250 ms
9.2.44	Selection abort time	maximum	200 µs
9.2.45	Selection time-out delay (note 2)	minimum	250 ms
9.2.48	System deskew delay	minimum	45 ns

Notes:
1 Cable Skew is measured at each SCSI device connection with the transmitted skew subtracted from the received skew.
2 This is a recommended time. It is not mandatory.

Table 31 - SCSI bus data & information phase ST timing values

Subclause	Timing description	Type	Timing values (note 4)				
			Asynch	Fast-5	Fast-10	Fast-20	Fast-40
9.2.2	ATN transmit setup time	min	90 ns	33 ns	33 ns	21,5 ns	19,25 ns
9.2.3	ATN receive setup time	min	45 ns	17 ns	17 ns	8,5 ns	6,75 ns
9.2.8	Cable skew (note 1)	max	4 ns	4 ns	4 ns	3 ns	2,5 ns
9.2.28	Receive assertion period (note 2)	min	N/A	70 ns	22 ns	11 ns	6,5 ns
9.2.29	Receive hold time (note 2 and note 3)	min	N/A	25 ns	25 ns	11,5 ns	4,75 ns
9.2.32	Receive negation period (note 2)	min	N/A	70 ns	22 ns	11 ns	6,5 ns
9.2.33	Receive setup time (note 2 and note 3)	min	N/A	15 ns	15 ns	6,5 ns	4,75 ns
9.2.28	Receive REQ (ACK) period tolerance	min	N/A	1,1 ns	1,1 ns	1,1 ns	1,1 ns
9.2.46	Signal timing skew	max	8 ns	8 ns	8 ns	5 ns	4,5 ns
9.2.39	REQ (ACK) period	nominal	N/A	200 ns	100 ns	50 ns	25 ns
9.2.52	Transmit assertion period (note 2)	min	N/A	80 ns	30 ns	15 ns	8 ns
9.2.53	Transmit hold time (note 2 and note 3)	min	N/A	53 ns	33 ns	16,5 ns	9,25 ns
9.2.55	Transmit negation period (note 2)	min	N/A	80 ns	30 ns	15 ns	8 ns
9.2.56	Transmit setup time (note 2 and note 3)	min	N/A	23 ns	23 ns	11,5 ns	9,25 ns
9.2.57	Transmit REQ (ACK) period tolerance	max	N/A	1 ns	1 ns	1 ns	1 ns

Notes:
1 Cable skew is measured at each SCSI device connection with the transmitted skew subtracted from the received skew.
2 See 9.3 for measurement points for the timing specifications.
3 See 9.4 for examples of how to calculate setup and hold timing.
4 SCSI bus timing values specified by the maximum transfer rate for the given range shall apply even if a slower transfer rate within the given range is negotiated.

Table 32 - SCSI bus data & information phase DT timing values

Subclause	Timing description	Type	Timing Values (note 4)				
			Fast-10	Fast-20	Fast-40	Fast-80	Fast-160
9.2.2	ATN transmit setup time	min	48,4 ns	29,2 ns	19,6 ns	14,8 ns	14 ns
9.2.3	ATN receive setup time	min	13,6 ns	7,8 ns	4,9 ns	3,45 ns	3 ns
9.2.8	Cable skew (note 1)	max	4 ns	3 ns	2,5 ns	2,5 ns	2,5 ns
9.2.11	Flow control receive hold time	min	N/A	N/A	N/A	N/A	3 ns
9.2.12	Flow control receive setup time	min	N/A	N/A	N/A	N/A	3 ns
9.2.13	Flow control transmit hold time	min	N/A	N/A	N/A	N/A	14 ns
9.2.14	Flow control transmit setup time	min	N/A	N/A	N/A	N/A	14 ns
9.2.15	pCRC receive hold time	min	11,6 ns	5,8 ns	2,9 ns	1,45 ns	N/A
9.2.16	pCRC receive setup time	min	18,6 ns	12,8 ns	9,9 ns	8,45 ns	N/A
9.2.17	pCRC transmit hold time	min	38,4 ns	19,2 ns	9,6 ns	4,8 ns	N/A
9.2.18	pCRC transmit setup time	min	48,4 ns	29,2 ns	19,6 ns	14,8 ns	N/A
9.2.28	Receive assertion period (note 2)	min	80 ns	40 ns	20 ns	8,5 ns	4,74 ns
9.2.29	Receive hold time (note 2 and note 3)	min	11,6 ns	5,8 ns	2,9 ns	1,45 ns	(0,08) ns
9.2.32	Receive negation period (note 2)	min	80 ns	40 ns	20 ns	8,5 ns	4,74 ns
9.2.33	Receive setup time (note 2 and note 3)	min	11,6 ns	5,8 ns	2,9 ns	1,45 ns	(6,33) ns
9.2.34	Receive REQ (ACK) period tolerance	min	0,7 ns	0,7 ns	0,7 ns	0,7 ns	0,06 ns
9.2.35	Receive REQ assertion period with P_CRCA transitioning	min	85,5 ns	48 ns	32,5 ns	21 ns	N/A
9.2.36	Receive REQ negation period with P_CRCA transitioning	min	85,5 ns	48 ns	32,5 ns	21 ns	N/A
9.2.46	Signal timing skew	max	26,8 ns	13,4 ns	6,7 ns	3,35 ns	4,85 ns
9.2.39	REQ (ACK) period	nominal	200 ns	100 ns	50 ns	25 ns	12,5 ns
9.2.52	Transmit assertion period (note 2)	min	92 ns	46 ns	23 ns	11,5 ns	5,69 ns
9.2.53	Transmit hold time (note 2 and note 3)	min	38,4 ns	19,2 ns	9,6 ns	4,8 ns	4,77 ns
9.2.55	Transmit negation period (note 2)	min	92 ns	46 ns	23 ns	11,5 ns	5,69 ns
9.2.56	Transmit setup time (note 2 and note 3)	min	38,4 ns	19,2 ns	9,6 ns	4,8 ns	(1,48) ns
9.2.57	Transmit REQ (ACK) period tolerance	max	0,6 ns	0,6 ns	0,6 ns	0,6 ns	0,06 ns
9.2.58	Transmit REQ assertion period with P_CRCA transitioning	min	97,5 ns	54 ns	35,5 ns	24 ns	N/A
9.2.59	Transmit REQ negation period with P_CRCA transitioning	min	97,5 ns	54 ns	35,5 ns	24 ns	N/A
9.2.54	Transmit ISI Compensation	max	note 5	note 5	note 5	note 5	1,0 ns
9.2.37	Receive Skew Compensation	max	N/A	N/A	N/A	N/A	4,4 ns
9.2.43	Residual Skew Error (note 6)	max	N/A	N/A	N/A	N/A	±0,15 ns
9.2.47	Strobe Offset Tolerance	max	N/A	N/A	N/A	N/A	±0,125 ns
9.2.30	Receive Internal Hold Time (note 6)	min	N/A	N/A	N/A	N/A	0,345 ns
9.2.31	Receive Internal Setup Time (note 6)	min	N/A	N/A	N/A	N/A	0,345 ns

Notes:
1 Cable skew is measured at each SCSI device connection with the transmitted skew subtracted from the received skew.
2 See 9.3 for measurement points for the timing specifications.
3 See 9.4 for examples of how to calculate setup and hold timing.
4 SCSI bus timing values specified by the maximum transfer rate for the given range shall apply even if a slower transfer rate within the given range is negotiated.
5 Calculated without any ISI compensation.
6 Calculated assuming timing budget shown in table 33.
7 Fast 160 SCSI devices shall not change timing parameters between training or reset events.

Table 33 - SCSI Fast-160 timing budget template

Item	Fast-160	Comments
Nominals:		
REQ(ACK) Period	12,5 ns	from table 32
Transfer period	6,25 ns	REQ/ACK period / 2
Ideal Setup/Hold	3,125 ns	REQ/ACK period / 4
Non-Compensatable (Early to Late):		**Worse case total of + and - time shift unless otherwise noted**
REQ(ACK) period tolerance / 2	0,06 ns	Tolerance of transmitter plus measurement error (note 1)
Driver time asymmetry	0,50 ns	
Receiver time asymmetry	0,35 ns	
System noise at launch	0,25 ns	Time impact
System noise at receiver	0,25 ns	Time impact
Near end Crosstalk	0,7 ns	Time impact
Chip noise in receiver	0,2 ns	Time impact
Receiver amplitude time skew	0,2 ns	With minimum signals
Clock jitter	0,25 ns	
Strobe offset tolerance	0,5 ns	Accuracy of centering strobe
Residual Skew error	0,3 ns	After skew compensation
Non-Compensatable total:	**3,56 ns**	
Compensatable:		**Worst Case**
Transmitter chip skew	0,75 ns	
Receiver chip skew	0,75 ns	
Cable skew	2,5 ns	
Two times trace skew	0,4 ns	Total for SCSI device pair
ISI of data	4,0 ns	Worse case pattern
ISI of REQ(ACK)	0,0 ns	Post preamble
May detect to shall detect ambiguity	0,0 ns	Assumed to be negligible in given chip
Compensatable total:	**8,4 ns**	
Assumed Compensation		
ISI Compensation	2,0 ns	Assumes 50% of ISI is compensated
Skew compensation	4,4 ns	Internal alignment of data signals to REQ or ACK
Compensation total:	**6,4 ns**	
Total Error Inputs:	**11,96 ns**	**Sum of compensatable and non-compensatable timings**
Post compensation error:	**5,56 ns**	**Total error inputs - compensation total**
Minimum compensated internal setup (int) data valid window	**0,345 ns**	**(transfer period - post compensation error) / 2 note 2**
Minimum compensated internal hold (int) data valid window	**0,345 ns**	**(transfer period - post compensation error) / 2 note 2**

Notes:
1 Tolerance adjusted for half cycle (transfer period)
2 Timing budgets in previous standards neglected asymmetry & detection ambiguity and lumps chip noise, clock jitter, cross-talk, noise, ISI and receiver amplitude skew into other terms (e.g., signal distortion skew) and/or ignores the effects.

9.2 Timing description

9.2.1 Arbitration delay

The minimum time a SCSI device shall wait from asserting the BSY signal for arbitration until the DATA BUS is examined to see if arbitration has been won (see 10.5). There is no maximum time.

9.2.2 ATN transmit setup time

The minimum time provided by the transmitter between the assertion of the ATN signal and the negation of the last ACK signal. When information unit transfers are enabled, the ATN transmit setup time is to the negation of the ACK signal corresponding to the last iuCRC transfer of an information unit.

Specified to provide the increased ATN receive setup time, subject to intersymbol interference, cable skew, and other distortions.

9.2.3 ATN receive setup time

~~The minimum time required at the receiver between the assertion of the ATN signal and the negation of the ACK signal to recognize the assertion of an Attention Condition.~~

The minimum time required at the receiver between the assertion of the ATN signal and the negation of the ACK signal to recognize the assertion of an attention condition. When information unit transfers are enabled, the ATN receive setup time is to the negation of the ACK signal corresponding to the last iuCRC transfer of an information unit.

Specified to ease receiver timing requirements.

~~NOTE 25 - Previous versions of this standard provided two system deskew delays of setup time.~~

9.2.4 Bus clear delay

The maximum time for a SCSI device to release all SCSI bus signals after:

a) the BUS FREE phase is detected (the BSY and SEL signals are both false for a bus settle delay);
b) the SEL signal is received from another SCSI device during the ARBITRATION phase;
c) the transition of the RST signal to true.

For item a) above, the maximum time for a SCSI device to release all SCSI bus signals is 1200 ns from the BSY and SEL signals first becoming both false. If a SCSI device requires more than a bus settle delay to detect BUS FREE phase, it shall release all SCSI bus signals within a bus clear delay minus the excess time.

9.2.5 Bus free delay

The minimum time that a SCSI device shall wait from its detection of the BUS FREE phase (BSY and SEL both false for a bus settle delay) until its assertion of the BSY signal in preparation for entering the ARBITRATION phase.

9.2.6 Bus set delay

The maximum time for a SCSI device to assert the BSY signal and its SCSI ID after it detects a BUS FREE phase for the purpose of entering the ARBITRATION phase.

9.2.7 Bus settle delay

The minimum time to wait for the bus to settle after changing certain control signals as called out in the protocol definitions.

Provides time for a signal transition to propagate from the driver to the terminator and back to the driver.

9.2.8 Cable skew

The maximum difference in propagation time allowed between any two SCSI bus signals measured between any two SCSI devices excluding any signal distortion skew delays.

9.2.9 Chip noise in receiver

The maximum transition to transition time shift due to the internal physics of the receiving SCSI device circuitry.

9.2.10 Clock jitter

The maximum transition to transition time shift of SCSI bus signals caused by short term variations in the transmitting SCSI device's clock.

9.2.11 Flow control receive hold time

The maximum time required by the initiator between the assertion of the REQ signal corresponding to the last iuCRC transfer of a SPI data streaming information unit and the changing of the P_CRCA signal.

Specified to ease receiver timing requirements.

9.2.12 Flow control receive setup time

The maximum time required by the initiator between the assertion of the P_CRCA signal and the assertion of the REQ signal corresponding to the last iuCRC transfer of a SPI data streaming information unit. Also, the maximum time required by the initiator between the negation of the P_CRCA signal and the assertion of the REQ signal corresponding to any valid data transfer of a SPI L_Q information unit.

Specified to ease receiver timing requirements.

9.2.13 Flow control transmit hold time

The minimum time provided by the target between the assertion of the REQ signal corresponding to the last iuCRC transfer of a SPI data streaming information unit and the changing of the P_CRCA signal.

Specified to provide the increased P_CRCA receive setup time, subject to intersymbol interference, cable skew, and other distortions.

9.2.14 Flow control transmit setup time

The minimum time provided by the target between the assertion of the P_CRCA signal and the assertion of the REQ signal corresponding to the last iuCRC transfer of a SPI data streaming information unit. Also, the minimum time provided by the target between the negation of the P_CRCA signal and the assertion of the REQ signal corresponding to any valid data transfer of a SPI L_Q information unit.

Specified to provide the increased P_CRCA receive setup time, subject to intersymbol interference, cable skew, and other distortions.

9.2.15 pCRC receive hold time

The minimum time required at the receiver between the transition of the REQ signal and the transition of the P_CRCA signal while pCRC protection is enabled (see 16.3.10).

9.2.16 pCRC receive setup time

The minimum time required at the receiver between the transition of the P_CRCA signal and the transition of the REQ signal while pCRC protection is enabled (see 16.3.10).

Specified to ease receiver timing requirements and ensure that this signal, which is outside CRC protection, is received correctly.

9.2.17 pCRC transmit hold time

The minimum time provided by the transmitter between the transition of the REQ signal and the transition of the P_CRCA signal while pCRC protection is enabled (see 16.3.10).

9.2.18 pCRC transmit setup time

The minimum time provided by the transmitter between the transition of the P_CRCA signal and the transition of the REQ signal while pCRC protection is enabled (see 16.3.10).

Specified to provide the increased receive setup time, subject to intersymbol interference, cable skew, and other distortions.

9.2.19 Data release delay

The maximum time for an initiator to release the DATA BUS, DB(P_CRCA), and/or DB(P1) signals, following the transition of the I/O signal from false to true.

9.2.20 DIFFSENS voltage filter time

The minimum time DIFFSENS voltage shall be sensed continuously within the voltage range of a valid SCSI bus mode.

~~9.2.21 Net Receive Internal Setup and Hold times~~

~~The net effective setup or hold time measured within the receiving SCSI device from the worse case bit (data or parity) to the compensated offset strobe. This time may not be observable to other than the SCSI device designer. Failure to meet the requirement may only be observable in terms of increased error rates.~~

9.2.22 Physical disconnection delay

The minimum time that a target shall wait after releasing BSY before participating in an ARBITRATION phase when honoring a DISCONNECT message from the initiator.

9.2.23 Power on to selection

The recommended maximum time from power application until a SCSI target is able to respond with appropriate status and sense data to the TEST UNIT READY, INQUIRY, and REQUEST SENSE commands (See SCSI Primary Commands-2 standard).

9.2.24 QAS arbitration delay

The minimum time a SCSI device with QAS enabled (see 16.3.10) shall wait from the detection of the MSG, C/D, and I/O signals being false to start QAS until the DATA BUS is examined to see if QAS has

been won (see 10.5).

9.2.25 QAS assertion delay

The maximum time allowed for a SCSI device to assert certain signals during QAS.

9.2.26 QAS release delay

The maximum time allowed for a SCSI device to release certain signals during QAS.

9.2.27 QAS non-DATA phase REQ (ACK) period

The minimum time a QAS-capable initiator shall ensure the REQ and ACK signals are asserted and that data is valid during COMMAND, MESSAGE, and STATUS phases.

9.2.28 Receive assertion period

The minimum time ~~required~~ provided at a SCSI device receiving a REQ signal for the signal to be asserted while using synchronous transfers, provided P_CRCA is not transitioning with pCRC protection enabled (see 16.3.10). Also, the minimum time required at a SCSI device receiving an ACK signal for the signal to be asserted while using synchronous transfers. For SE fast-5 and fast-10 operation, the time period is measured at the 0,8 V level. For SE fast-20 operation the period is measured at the 1,0 V level. For LVD see figure 48 and figure 49 for signal measurement points.

9.2.29 Receive hold time

For ST data transfers the minimum time ~~required~~ provided at the receiving SCSI device between the assertion of the REQ signal or the ACK signals and the changing of the DATA BUS, DB(P_CRCA), and/or DB(P1) signals while using synchronous transfers, provided P_CRCA is not transitioning with pCRC protection enabled (see 16.3.10). For DT data transfers the minimum time required at the receiving SCSI device between the transition (i.e. assertion or negation) of the REQ signal or the ACK signals and the changing of the DATA BUS, DB(P_CRCA), and/or DB(P1) signals while using synchronous transfers.

9.2.30 Receive internal hold time

The minimum time provided for hold time in the receive data detector after allowance for timing errors and timing compensation from all sources measured from the worse case bit (i.e., data or parity) to the compensated offset strobe.

NOTE 26 - This time may not be observable to other than the SCSI device designer.

9.2.31 Receive internal setup time

The minimum time provided for setup time in the receive data detector after allowance for timing errors and timing compensation from all sources measured from the worse case bit (i.e., data or parity) to the compensated offset strobe.

NOTE 27 - This time may not be observable to other than the SCSI device designer.

9.2.32 Receive negation period

The minimum time required at a SCSI device receiving a REQ signal for the signal to be negated while using synchronous transfers. Also, the minimum time required at a SCSI device receiving an ACK signal for the signal to be asserted while using synchronous transfers. For SE fast-5 and fast-10 operation, the time period is measured at the 2,0 V level. For SE fast-20 operation the period is measured at the 1,9 V level. For LVD see figure 48 and figure 49 for signal measurement points.

9.2.33 Receive setup time

For ST data transfers the minimum time ~~required~~ provided at the receiving SCSI device between the changing of DATA BUS, DB(P_CRCA), and/or DB(P1) signals and the assertion of the REQ signal or the ACK signal while using synchronous transfers. For DT data transfers the minimum time required at the receiving SCSI device between the changing of DATA BUS, DB(P_CRCA), and/or DB(P1) signals and the transition of the REQ signal or the ACK signal while using synchronous transfers.

9.2.34 Receive REQ (ACK) period tolerance

The minimum tolerance that a SCSI device shall allow to be subtracted from the REQ (ACK) period. The tolerance comprises the transmit REQ (ACK) tolerance plus a measurement error due to noise.

9.2.35 Receive REQ assertion period with P_CRCA transitioning

The minimum time required at a SCSI device receiving a REQ signal for the signal to be asserted while using synchronous transfers with P_CRCA transitioning with pCRC protection enabled (see 16.3.10).

Specified to ensure that the assertion period is longer than the receive hold time plus the receive setup time.

9.2.36 Receive REQ negation period with P_CRCA transitioning

The minimum time required at a SCSI device receiving a REQ signal for the signal to be negated while using synchronous transfers with P_CRCA transitioning with pCRC protection enabled (see 16.3.10).

Specified to ensure that the negation period is longer than the receive hold time plus the receive setup time.

9.2.37 Receive Skew Compensation

The effective reduction in worse case timing skew of data, parity, and strobe signals provided by the receiving SCSI device but not directly observable at the receiving SCSI device connector.

9.2.38 Receiver amplitude time skew

The maximum time shift of SCSI bus signals caused by the difference in receiver switching delay of a minimum amplitude signal versus a maximum amplitude signal.

9.2.39 REQ (ACK) period

The REQ (ACK) period during synchronous transfers, specified in table 32 for ST DATA phases and in table 32 for DT DATA phases, is the nominal time between adjacent assertion edges of the REQ or ACK signal for the fastest negotiated data transfer rate. For the purpose of calculating the actual REQ (ACK) period tolerance the REQ (ACK) period should be measured without interruptions (e.g., offsets pauses). To minimize the impact of cross-talk and ISI the measurements should be made by averaging the time between edges during long (i.e., greater than 512 bytes) all zero or all ones data transfers and by ignoring the first and last 10 transitions.

In DT DATA phases the negotiated transfer period for data is half of the REQ (ACK) period since data is qualified on both the assertion and negation edges of the REQ or ACK signal. In ST DATA phases the negotiated transfer period for data is equal to the REQ (ACK) period during synchronous transfers since data is only qualified on the assertion edge of the REQ or ACK signal.

9.2.40 Reset delay

The minimum time that the RST signal shall be continuously true before the SCSI device shall initiate a

reset.

9.2.41 Reset hold time

The minimum time that the RST signal is asserted. There is no maximum time.

9.2.42 Reset to selection

The recommended maximum time from after a reset condition until a SCSI target is able to respond with appropriate status and sense data to the TEST UNIT READY, INQUIRY, and REQUEST SENSE commands (See SCSI Primary Commands-2 standard).

9.2.43 Residual skew error

The maximum timing error between the deskewed data and REQ or ACK internal to the receiving SCSI device after skew compensation.

9.2.44 Selection abort time

The maximum time that a SCSI device shall take from its most recent detection of being selected or reselected until asserting the BSY signal in response. This time-out is required to ensure that a target or initiator does not assert the BSY signal after a SELECTION or RESELECTION phase has been aborted.

9.2.45 Selection time-out delay

The minimum time that an initiator or target should wait for the assertion of the BSY signal during the SELECTION or RESELECTION phase before starting the time-out procedure. Note that this is only a recommended time period.

9.2.46 Signal timing skew

The maximum signal timing skew occurs when transferring random data and in combination with interruptions of the REQ or ACK signal transitions (e.g., pauses caused by offsets). The signal timing skew includes cable skew (measured with 0101... patterns) and signal distortion skew caused by random data patterns and transmission line reflections as shown in figure 46, figure 47, figure 48, and figure 49.

The receiver detection range is the part of the signal between the "may detect" level and the "shall detect" level on either edge. (see 9.3)

NOTE 28 - For timing budget purposes the value stated in ~~the table~~ table 32 is calculated without the benefit of skew compensation.

9.2.47 Strobe offset tolerance

~~The tolerance on the time used to delay the compensated REQ/ACK to strobe the data and parity signals provided by the receiving SCSI device but not directly observable at the receiving SCSI device connector.~~

The time tolerance of centering the compensated REQ or ACK strobe in the transfer period during the training pattern.

9.2.48 System deskew delay

The minimum time that a SCSI device should wait after receiving a SCSI signal to ensure that any signals transmitted at the same time are valid. The system deskew delay shall not be applied to the synchronous transfers.

9.2.49 System noise at launch

The maximum time shift of SCSI bus signals caused by system noise at the transmitter (e.g., noise caused by current changes in the voice coil) measured at the transmitting SCSI device connector.

9.2.50 System noise at receiver

The maximum time shift of SCSI bus signals caused by system noise at the receiver (e.g., noise caused by current changes in the voice coil) measured at the receiving SCSI device connector not including the time shift from the system noise at launch.

9.2.51 Time asymmetry

The maximum time difference between the asserted and negated signal for data, REQ, or ACK transitions that are intended to be equidistant.

9.2.52 Transmit assertion period

The minimum time that a target shall assert the REQ signal while using synchronous transfers, provided it is not transitioning P_CRCA with pCRC protection enabled (see 16.3.10). Also, the minimum time that an initiator shall assert the ACK signal while using synchronous transfers.

9.2.53 Transmit hold time

For ST data transfers the minimum time provided by the transmitting SCSI device between the assertion of the REQ signal or the ACK signal and the changing of the DATA BUS, DB(P_CRCA), and/or DB(P1) signals while using synchronous transfers. For DT data transfers the minimum time provided by the transmitting SCSI device between the transition of the REQ signal or the ACK signal and the changing of the DATA BUS, DB(P_CRCA), and/or DB(P1) signals while using synchronous transfers.

9.2.54 Transmit ISI ~~Precompensation~~ Compensation

The effective reduction in worse case ISI timing shift provided by the transmitting SCSI device as seen at the receiving SCSI device connector.

9.2.55 Transmit negation period

The minimum time that a target shall negate the REQ signal while using synchronous transfers, provided it is not transitioning P_CRCA with pCRC protection enabled (see 16.3.10). Also, the minimum time that an initiator shall negate the ACK signal while using synchronous transfers.

9.2.56 Transmit setup time

For ST data transfers the minimum time provided by the transmitting SCSI device between the changing of DATA BUS, DB(P_CRCA), and/or DB(P1) signals and the assertion of the REQ signal or the ACK signal while using synchronous transfers. For DT data transfers the minimum time provided by the transmitting SCSI device between the changing of DATA BUS, DB(P_CRCA), and/or DB(P1) signals and the transition of the REQ signal or the ACK signal while using synchronous transfers.

9.2.57 Transmit REQ (ACK) period tolerance

The maximum tolerance that a SCSI device may subtract from the REQ (ACK) period.

9.2.58 Transmit REQ assertion period with P_CRCA transitioning

The minimum time that a target shall assert the REQ signal during a synchronous transfer DT DATA phase while transitioning P_CRCA with pCRC protection enabled (see 16.3.10).

Specified to provide the increased receive REQ assertion period, subject to loss on the interconnect.

9.2.59 Transmit REQ negation period with P_CRCA transitioning

The minimum time that a target shall negate the REQ signal during a synchronous transfer DT DATA phase while transitioning P_CRCA with pCRC protection enabled (see 16.3.10).

Specified to provide the increased receive REQ negation period, subject to loss on the interconnect.

9.3 Measurement points

9.3.1 Measurement points overview

The measurements points for SE and LVD ACK, REQ, DATA, P_CRCA, and PARITY signals are defined in the following subclauses. All measurements are at the SCSI connector.

When paced transfers are enabled the timing shall be measured relative to the zero crossing of the differential signal.

9.3.2 SE fast-5 and fast-10 measurement points

SE SCSI devices with data transfer rates up to and including fast-10 shall use the measurement points defined in figure 46 for the measurement of the timing values. The rise and fall times for the SE REQ/ACK signals shall be nominally the same as for the SE DATA, DB(P_CRCA), and DB(P1) signals.

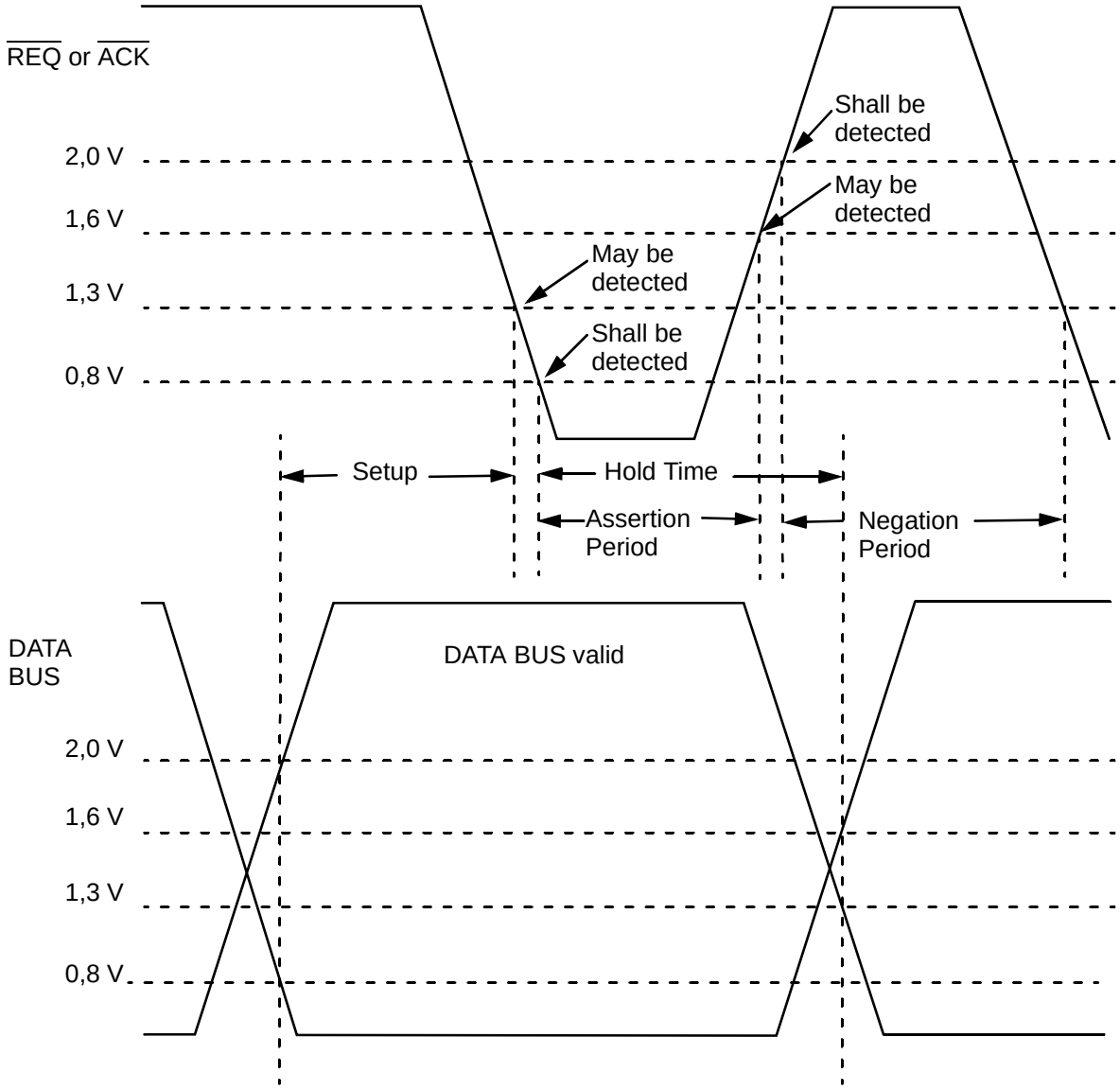


Figure 46 - Fast-5 and fast-10 SE timing measurement points

9.3.3 SE fast-20 measurement points

SE SCSI devices with data transfer rates up to and including fast-20 shall use the measurement points defined in figure 47 for the measurement of the timing values. The rise and fall times for the SE REQ and ACK signals shall be nominally the same as for the SE DATA, DB(P_CRCA), and DB(P1) signals.

SE fast-20 timing measurement points shall apply even if a slower transfer rate is negotiated.

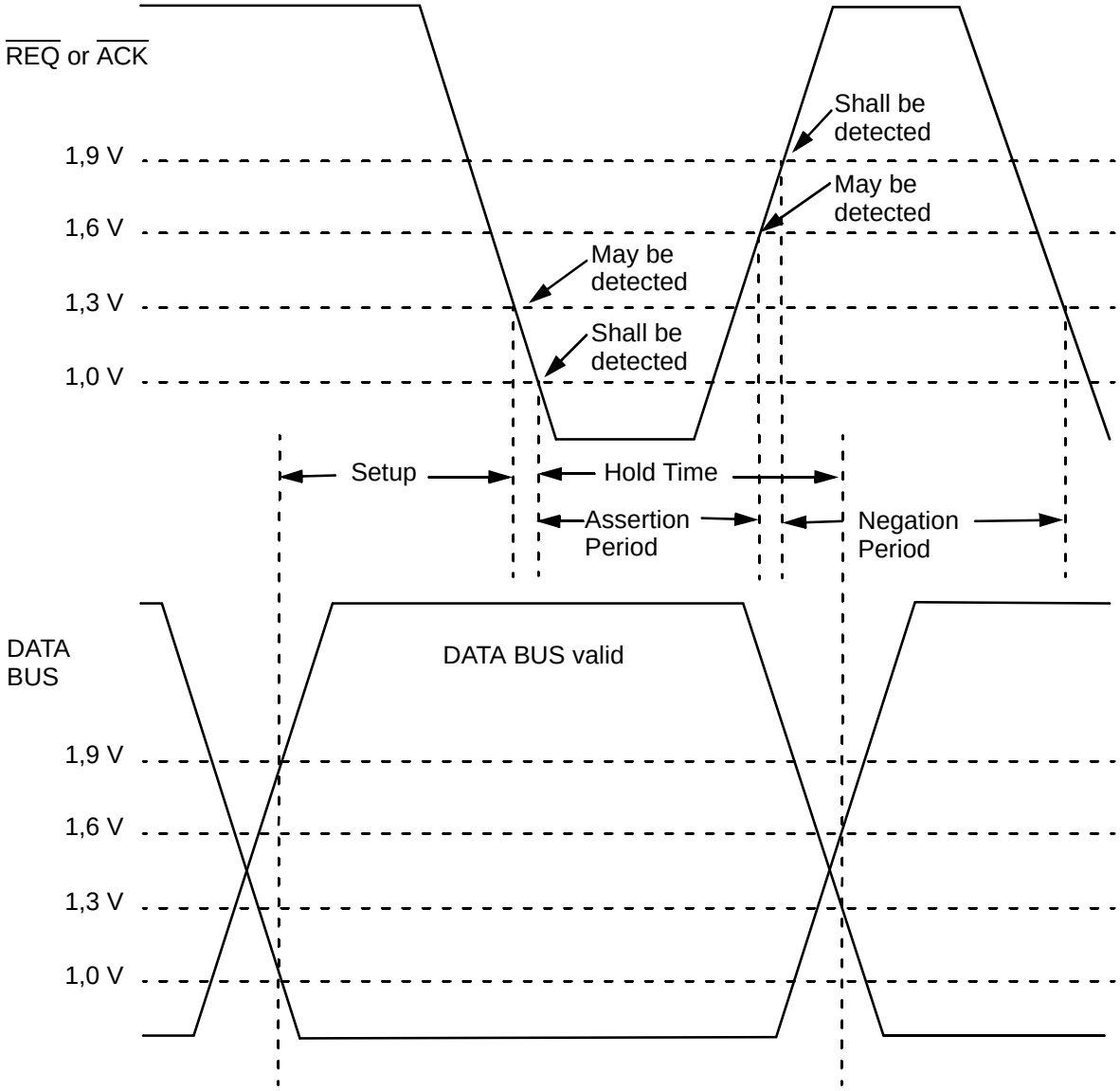
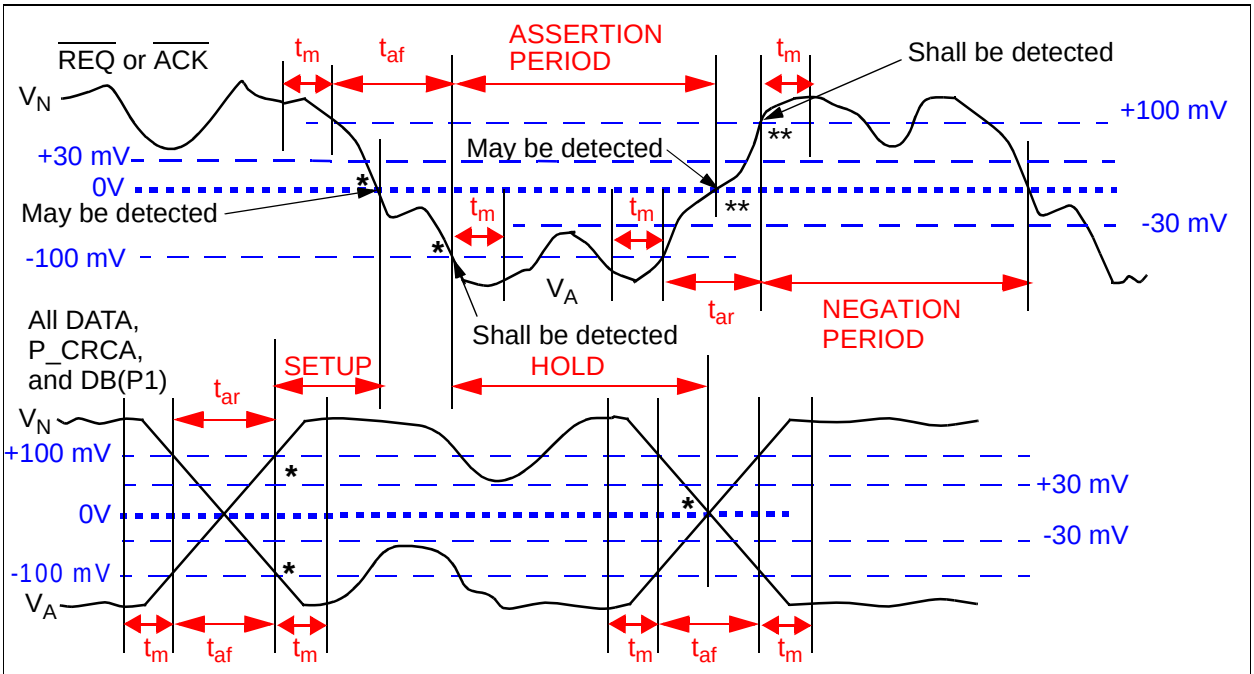


Figure 47 - Fast-20 SE timing measurement points

9.3.4 LVD measurement points

When transferring data using ST DATA phases LVD SCSI devices shall use the measurement points defined in figure 48 for the measurement of the timing values. When transferring data using DT DATA phases LVD SCSI devices with paced transfers disabled shall use the measurement points defined in figure 49 for the measurement of the timing values. When transferring data using DT DATA phases LVD SCSI devices with paced transfers enabled shall use the measurement points defined in figure 50 for the measurement of the timing values. The rise and fall times for the LVD REQ and ACK signals shall be nominally the same as for the LVD DATA, P_CRCA, and DB(P1) signals.



* Use the crossing that yields the shorter SETUP and HOLD time

** Use the crossing that yields the shorter ASSERTION PERIOD and NEGATION PERIOD

Notes:

1 V_N - negated signal

2 V_A - asserted signal

3 t_m = 1,25 ns minimum

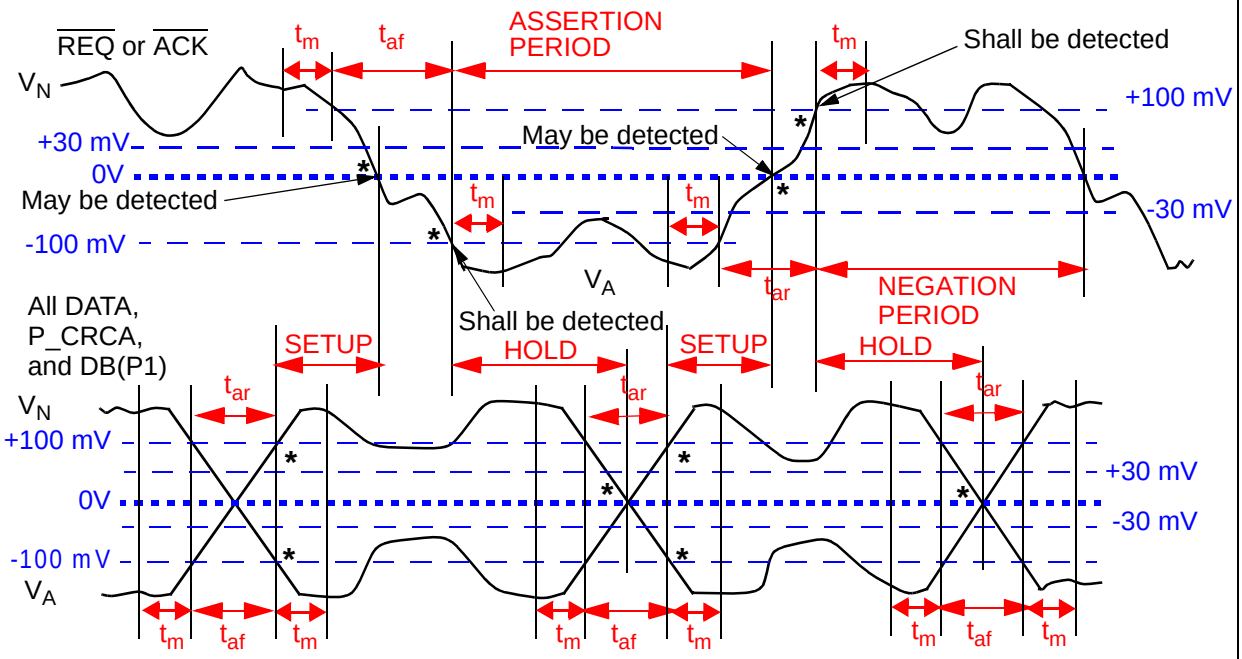
4 V_A or V_N are required to drive the 100 mV at the leading edge of the transition. The absolute value of those signals shall be at least 100 mV for at least t_m before and after the transition.

5 Differential voltage signals in all cases.

6 t_{af} and t_{ar} shall be less than 3 ns.

7 Any signal structure may occur at the receiver while in the t_{af} or t_{ar} region including slope reversal.

Figure 48 - LVD timing measurement points for ST data transfers



* Use the crossing that yields the shorter SETUP time, HOLD time, ASSERTION PERIOD, and NEGATION PERIOD

Notes:

1 V_N - negated signal

2 V_A - asserted signal

3 t_m = 1,25 ns minimum

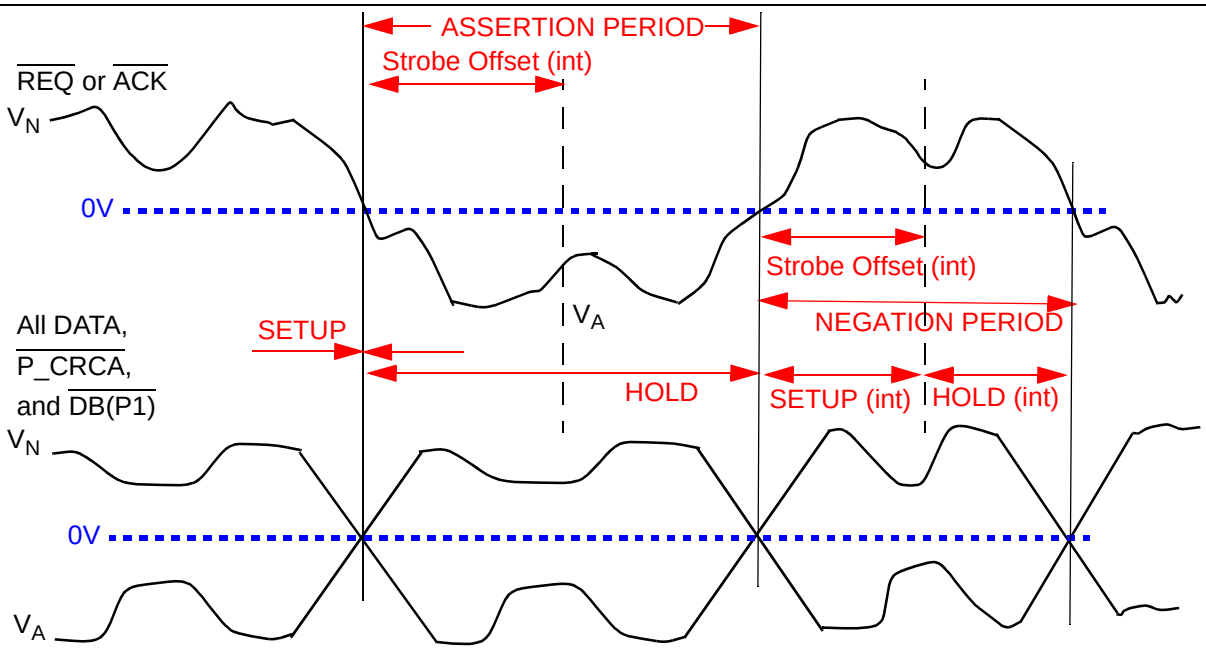
4 V_A or V_N are required to drive the 100 mV at the leading edge of the transition. The absolute value of those signals shall be at least 100 mV for at least t_m before and after the transition.

5 Differential voltage signals in all cases.

6 t_{af} and t_{ar} shall be less than 3 ns.

7 Any signal structure may occur at the receiver while in the t_{af} or t_{ar} region including slope reversal.

Figure 49 - LVD timing measurement points for DT data transfers (paced transfers disabled)



Notes:
1 V_N - negated signal
2 V_A - asserted signal
3 int = A value that is not observable at the SCSI connector.
4 Differential voltage signals in all cases.
5 Strobe Offset is from the actual detected time to the implemented offset time. Since effective timing is based upon actual detection, all timing measurements shall be based upon zero crossings for all paced transfers.

Figure 50 - LVD timing measurement points for DT data transfers (paced transfers enabled)

~~The signal~~ Non-isolated transitions and clocking signals shall transition from -~~TBD~~100 to +~~TBD~~100 mV or +~~TBD to TBD~~100 to -100 mV in 0 to 3 ns, the waveform between -~~TBD~~100 and +~~TBD~~100 mV is not otherwise specified. The absolute value of the signals shall remain above the ~~TBD~~100 mV level for 1,25 ns at each end of the transition. The absolute value of the signals shall not drop below ~~TBD~~30 mV except during the transitions (see 7.3.2).

Isolated transitions shall transition from at least 200 mV and shall be a minimum of 50 mV from the zero crossing (e.g., a -300 mV level shall transition to -50 mV for a valid transition).

Driver timing parameters applied at the connector of the transmitting SCSI device are defined in figure 48, figure 49, and figure 50. The driver timing parameters shall be measured using the circuit and test conditions defined in A.2.6. Receiver timing parameters applied at the connector of the receiving SCSI device are defined in figure 48 and figure 49. The receiver timing parameters include the effects of arbitrary data patterns and REQ/ACK pauses.

Figure 51 shows the LVD signal requirements at the receiving SCSI device with synchronous transfers. During paced transfers with precompensation enabled SCSI devices shall operate with signals at the receiving SCSI device meeting ~~either figure 52, figure 54 or both. Mask 1 is applicable to signals that have more timing margin than those for mask 2 and allows less amplitude margin than does mask 2. The~~ xxx. ~~amplitude margin of mask 1 may result in timing margin loss internal to the receiver. The higher amplitude~~

~~margin of mask 2 should result in less timing margin loss internal to the receiver.~~

During paced transfers noncompensated data signals shall operate with signals at the receiving SCSI device meeting figure 52, figure 53, and figure 54.

Editors Note 3 - GOP: There needs to be figure added to show the minimum mask when precomp is ~~disabled~~enabled.

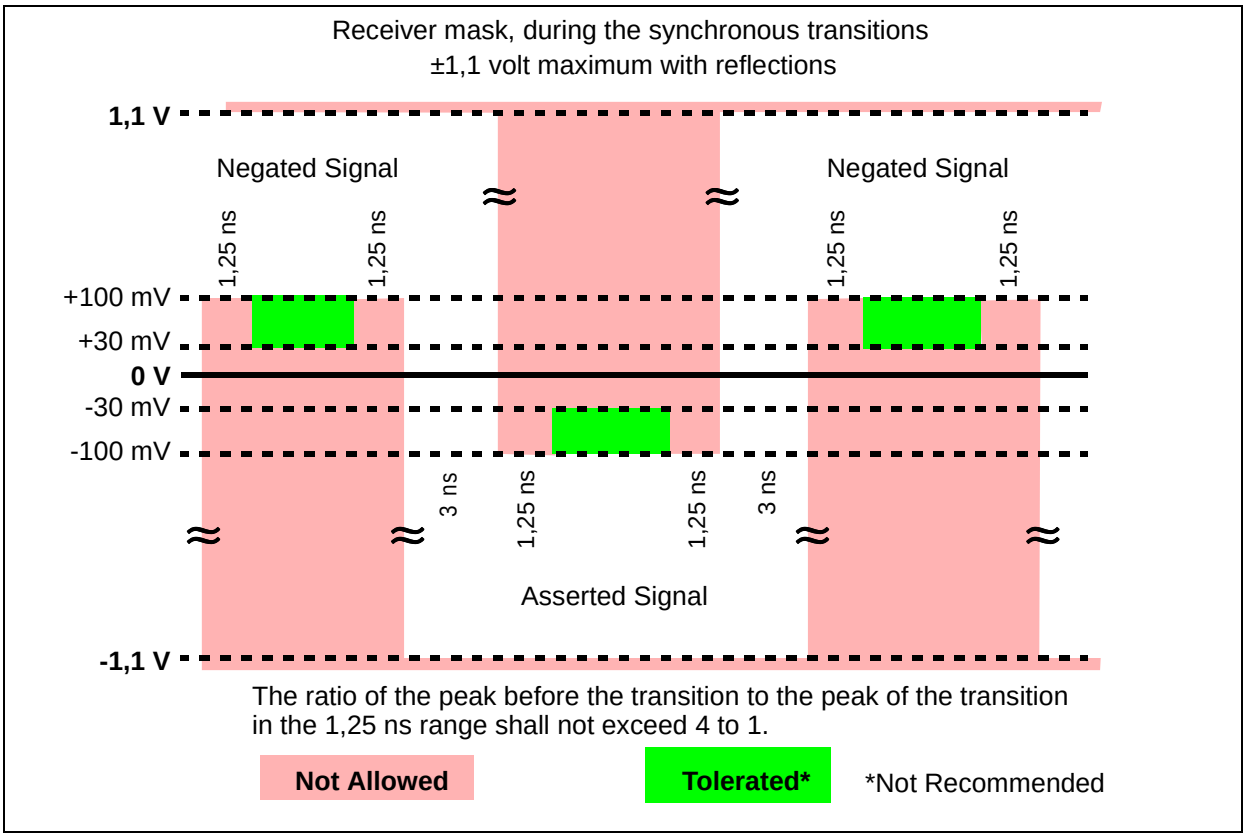


Figure 51 - LVD receiver mask (synchronous transfers)

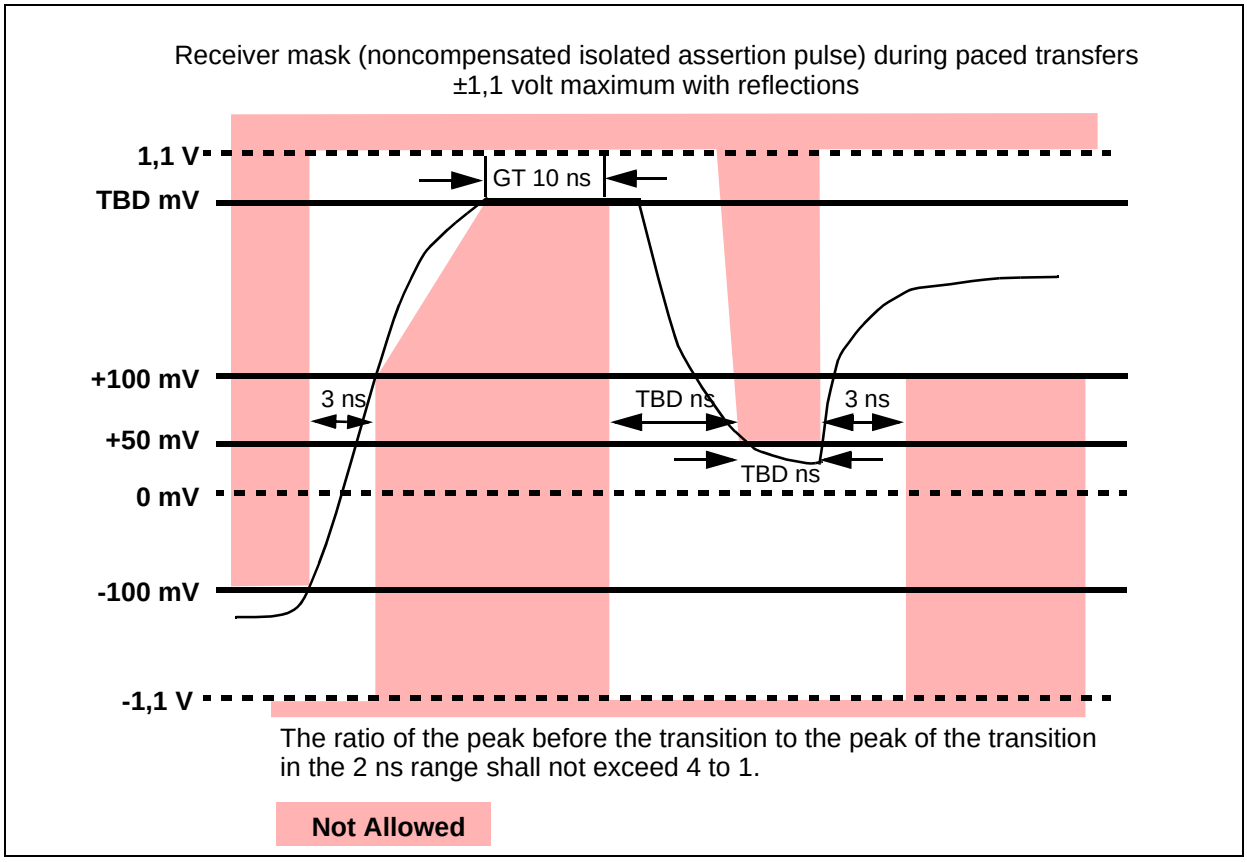


Figure 52 - LVD receiver mask ~~1~~ for noncompensated isolated negation pulse (paced transfers)

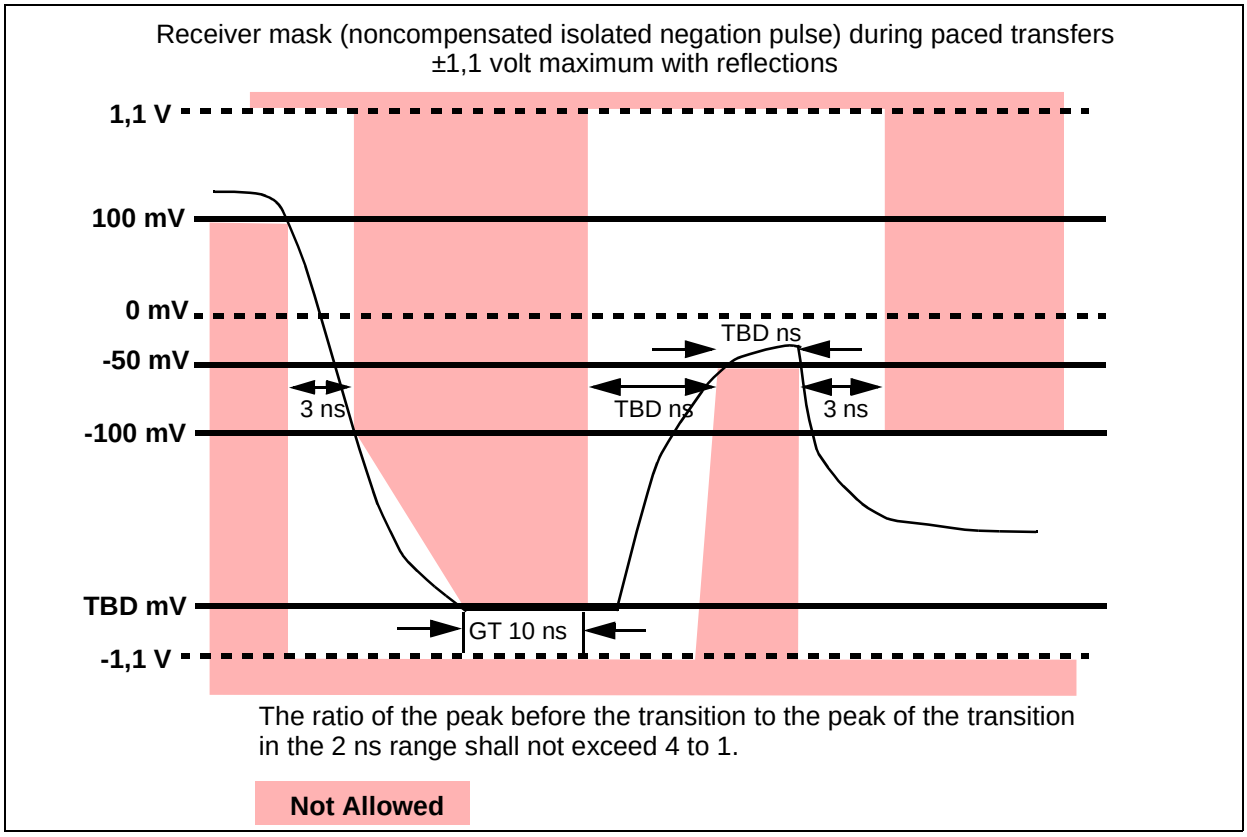


Figure 53 - LVD receiver mask ~~2~~ for noncompensated isolated negation pulse (paced transfers)

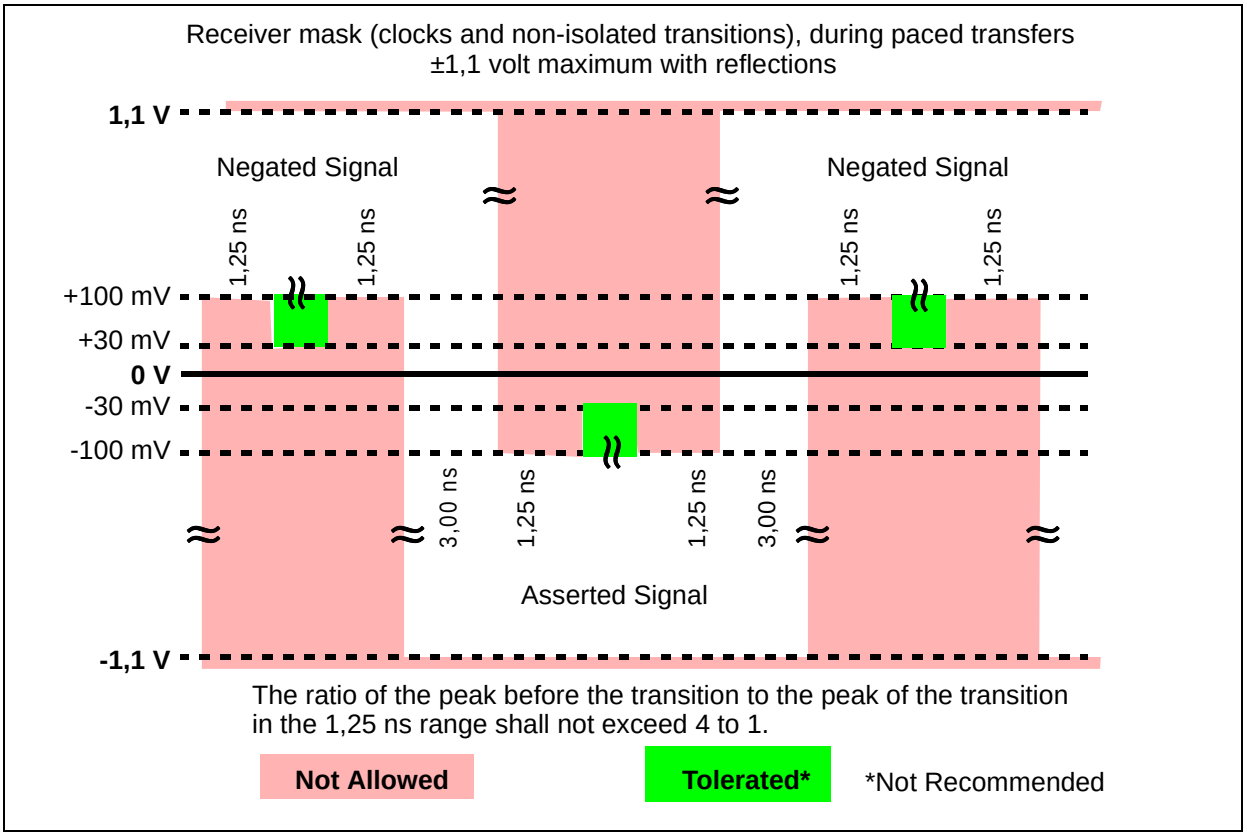


Figure 54 - LVD receiver mask for clocks and non-isolated transitions (paced transfers)

9.4 Setup and hold timings

9.4.1 ST data transfer calculations

Figure 55 shows how the setup and hold times are calculated for various physical configurations on SCSI devices that support ST data transfers. The minimum set up and hold timings specified in figure 55 shall be used. Note that these values are different for the driver and the receiver but that the receiver sensitivity provides the threshold points for both. This is required because both extreme cases of attenuation need to be covered:

a) receivers connected to drivers with very short interconnect, and
b) receivers connected to drivers through worst case interconnect.

Table 32 specifies setup and hold times at the device connector. Figure 55 illustrates a possible timing budget behind the device connector, with time apportioned to board skew and to the protocol chip.

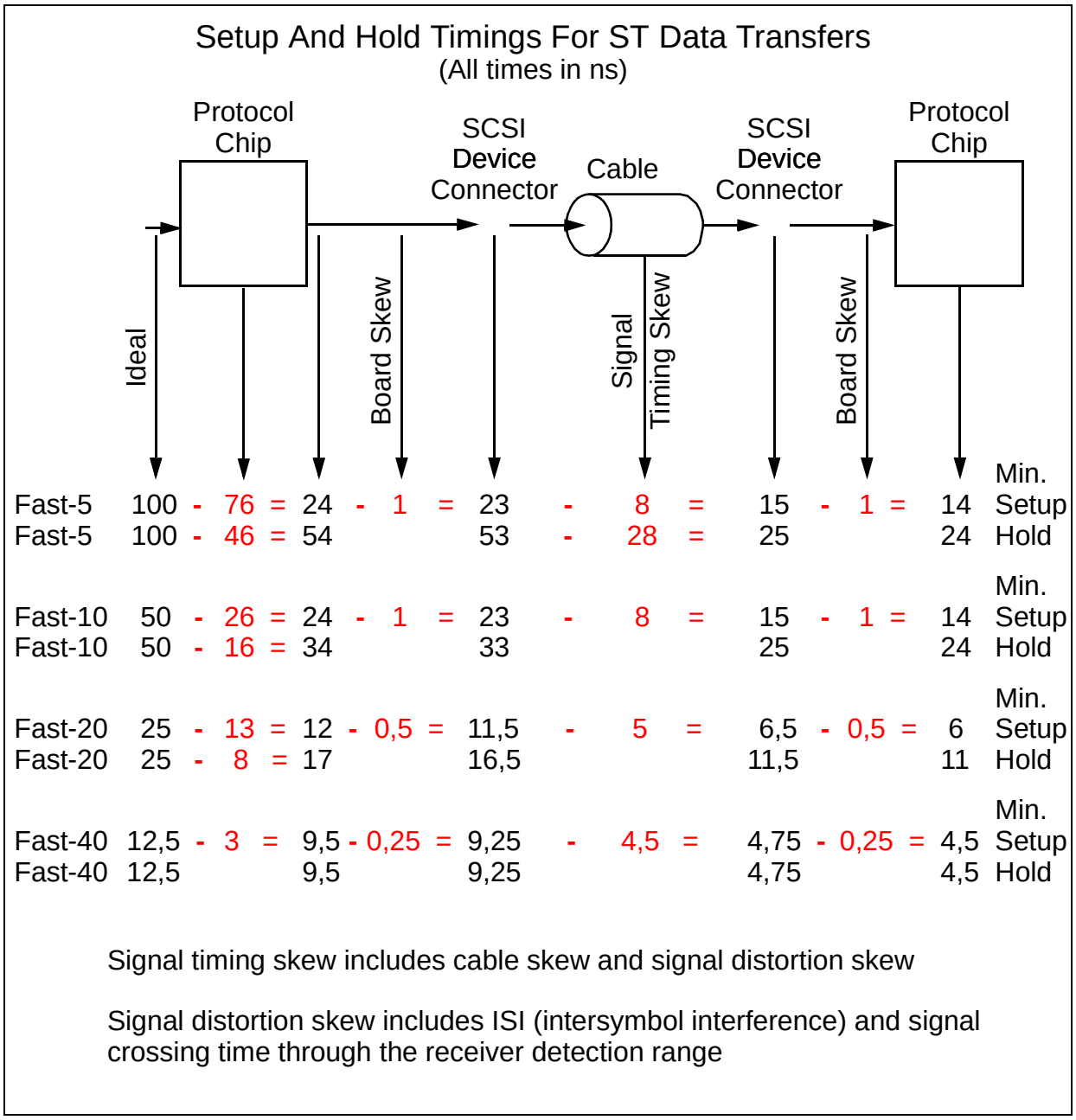


Figure 55 - System setup and hold timings for ST data transfers (all times in ns)

9.4.2 DT data transfer calculations

Figure 56 shows how the setup and hold times are calculated for various physical configurations using synchronous transfers on SCSI devices that support DT data transfers. Figure 57 and figure 58 show how the setup and hold times are calculated for various physical configurations using paced transfers. The minimum set up and hold timings specified in figure 56, figure 57, and figure 58 shall be used. Note that these values are different for the driver and the receiver but that the receiver sensitivity provides the threshold points for both. This is required because both extreme cases of attenuation need to be covered:

a) receivers connected to drivers with very short interconnect, and
b) receivers connected to drivers through worst case interconnect.

Table 32 specifies setup and hold times at the device connector. Figure 56, figure 57 and ~~figure 57~~figure 58 ~~illustrates a~~ illustrate possible timing ~~budget~~ budgets behind the device connector, with time apportioned to board skew and to the protocol chip.

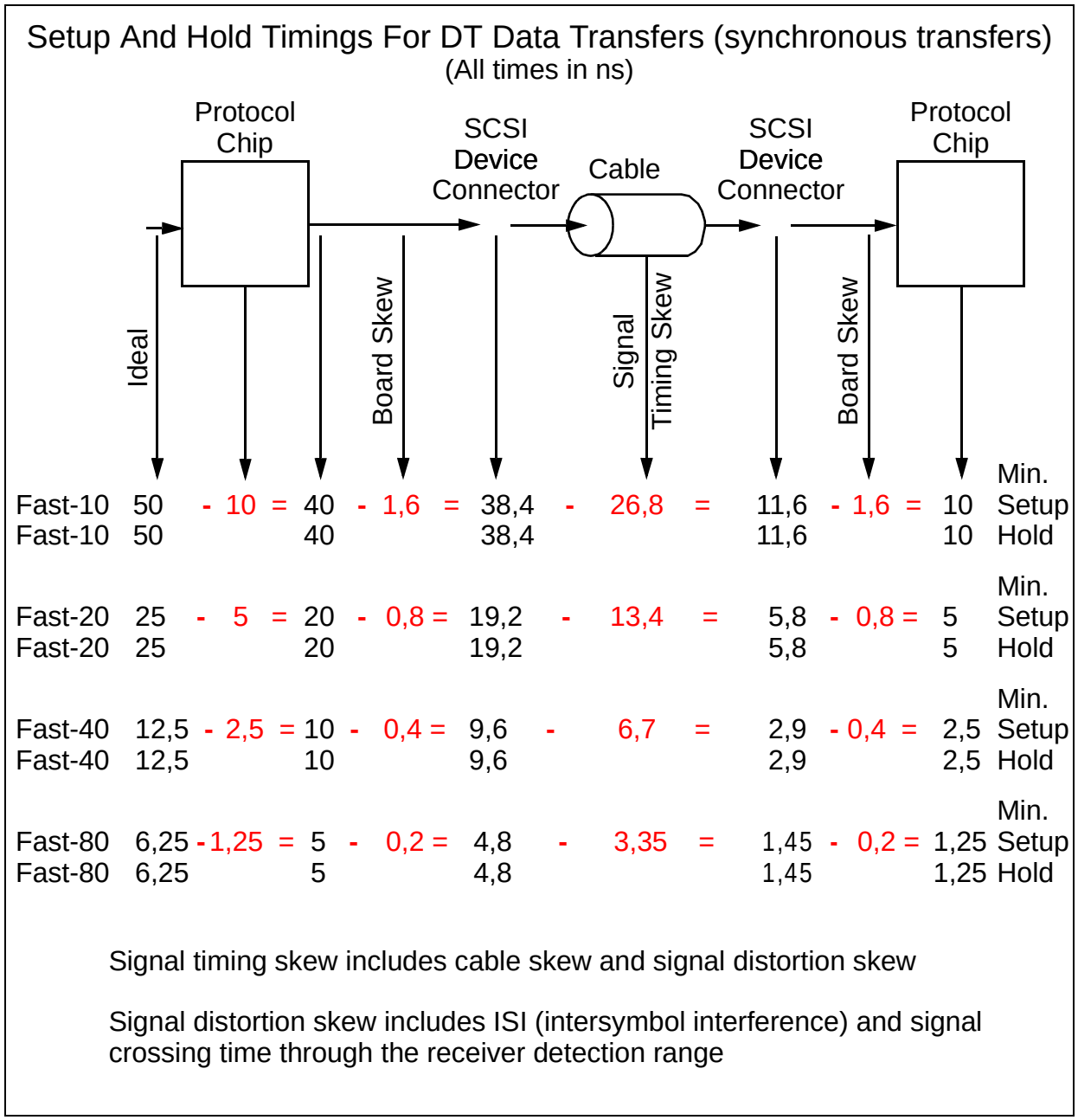
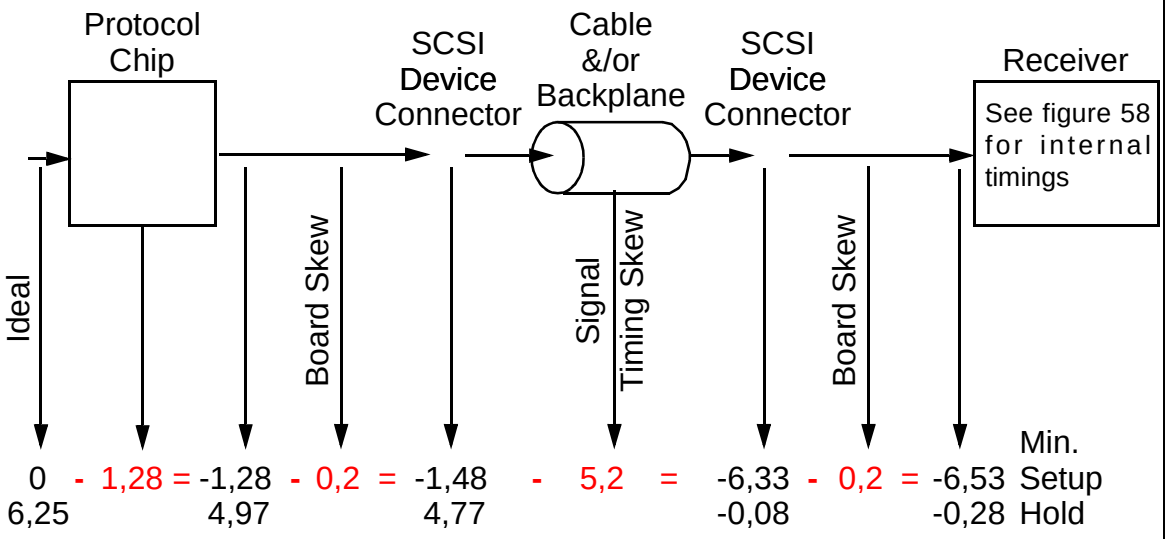


Figure 56 - System setup and hold timings for DT data transfers (synchronous transfers)

Setup And Hold Timings For Fast-160 DT Data Transfers (paced transfers)
(All times in ns)



Notes:
1 The transmitting SCSI device may reduce the timing margin by the following worse case factors: bit period tolerance (i.e., REQ(ACK) Period Tolerance / 2); driver asymmetry; transmitter chip skew, clock jitter; system noise at launch; and board skew.
2 The transmission of data from one SCSI device to another may have the timing margin reduced by the signal timing skew that comprises the following worse case factors; crosstalk; cable skew; and ISI. If precompensation is enabled the worse case signal timing skew is reduced by the ISI compensation value. Portions of the crosstalk and ISI may also be observable at the transmitting device connector. Consequently transmit setup and transmit hold time should be measured with an alternating pattern at the negotiated data rate with all data signals other than that being measured held at a continual assertion or negation.
3 The reception of data within a SCSI device may have the timing margin reduced or increased by the following additional worse case factors: board skew; receiver amplitude time skew; receiver chip skew; receiver time asymmetry; system noise at the receiver; receiver chip noise applied at receiver due to small signal physics; ISI compensation; skew compensation; strobe offset tolerance; and residual skew error.

Figure 57 - System setup and hold timings for fast-160 DT data transfers (paced transfers)

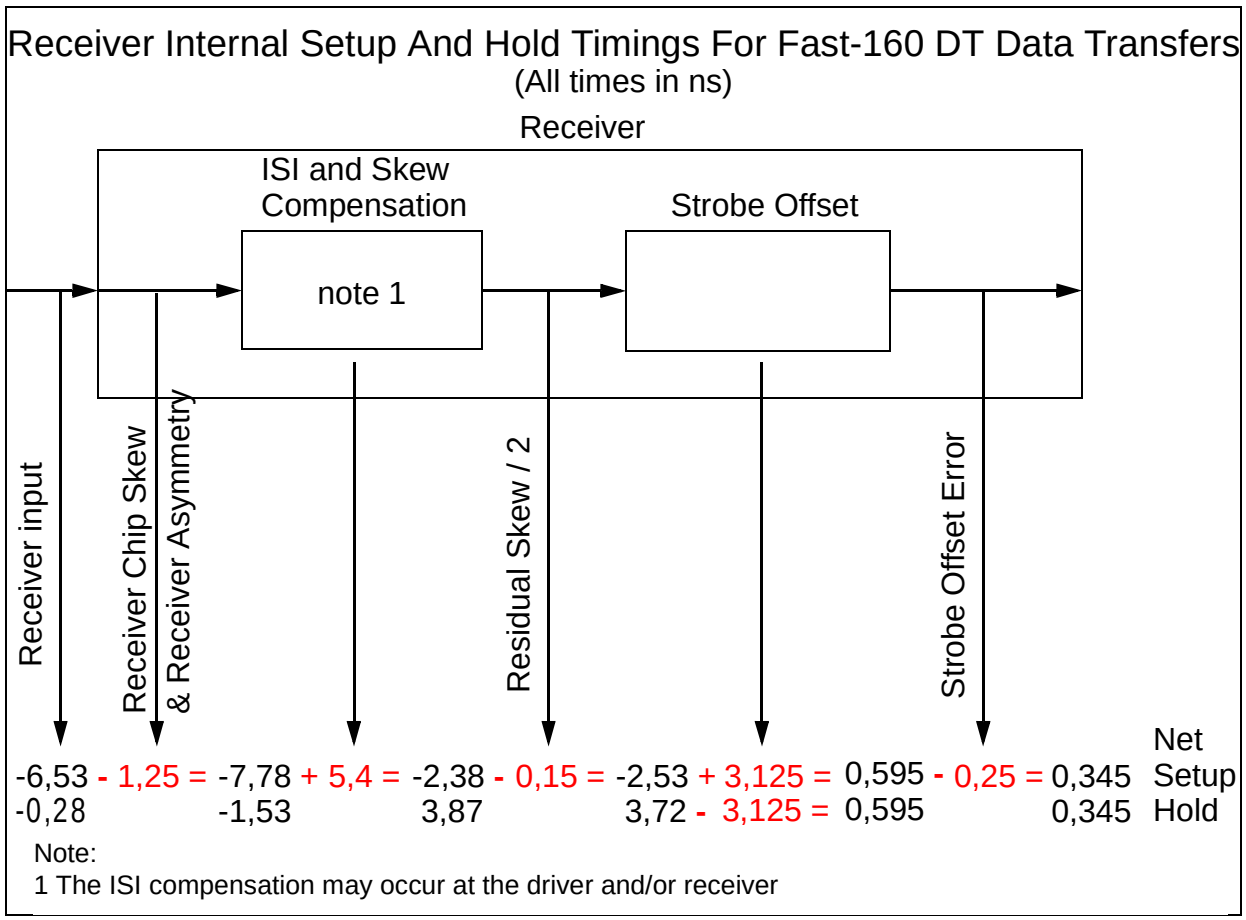


Figure 58 - ~~System~~ Receiver internal setup and hold timings for fast-160 DT data ~~transfers (paced transfers)~~

10 SCSI bus phases

10.1 SCSI bus phases overview

The SCSI architecture includes the following phases:

a) BUS FREE phase,
b) ARBITRATION phase,
c) SELECTION phase,
d) RESELECTION phase,
e) COMMAND phase,
f) DATA phase,
g) STATUS phase, and
h) MESSAGE phase.

The COMMAND phase, DATA phase, STATUS phase, and MESSAGE phase are collectively termed the information transfer phases.

The SCSI bus phases are defined such that the SCSI bus is never in more than one phase at any given time. In the following descriptions, signals that are not mentioned shall not be asserted.

10.2 BUS FREE phase

The BUS FREE phase indicates that there is no current task and that the SCSI bus is available for a physical connection or physical reconnection.

SCSI devices shall detect the BUS FREE phase after the SEL and BSY signals are both false for at least one bus settle delay.

SCSI devices shall release all SCSI bus signals within one bus clear delay after the BSY and SEL signals become continuously false for one bus settle delay. If a SCSI device requires more than one bus settle delay to detect the BUS FREE phase then it shall release all SCSI bus signals within one bus clear delay minus the excess time to detect the BUS FREE phase. The total time to clear the SCSI bus shall not exceed one bus settle delay plus one bus clear delay.

During normal operation the BUS FREE phase is entered when a target releases the BSY signal.

10.3 Unexpected bus free phases

An unexpected bus free occurs when an initiator detects a BUS FREE phase that is not expected. Initiators shall expect a BUS FREE phase to occur after one of the following occurs:

a) after a hard reset is detected;
b) after an ABORT TASK task management function is successfully received by a target;
c) after an ABORT TASK SET task management function is successfully received by a target;
d) after a CLEAR TASK SET task management function is successfully received by a target;
e) after a LOGICAL UNIT RESET task management function is successfully received by a target;
f) after a TARGET RESET task management function is successfully received by a target;
g) after a CLEAR ACA task management function is successfully received by a target;
h) after a DISCONNECT message is successfully transmitted from a target;
i) after a TASK COMPLETE message is successfully transmitted from a target;
j) after the release of the SEL signal after a SELECTION or RESELECTION phase time-out;
k) after a transceiver mode change:
l) after a PPR negotiation in response to a selection using attention condition when information unit transfers are enabled (see 16.3.10): or

m) after any successful message negotiation that causes information unit transfers to be enabled (see 16.3.10) or disabled (see 16.3.10).

The target uses an unexpected bus free to inform the initiator of a protocol error. The target may switch to a BUS FREE phase at any time, except during an ARBITRATION phase, independent of any attention condition.

The target shall terminate the task that was the current task before the BUS FREE phase by clearing all data and status for that task. The target may optionally prepare sense data that may be retrieved by a REQUEST SENSE command. However, an unexpected bus free shall not create an exception condition.

The initiator shall terminate the task that was the current task before the BUS FREE phase occurred and shall manage this condition as an exception condition.

10.4 Expected bus free phases

Initiators may expect a bus free to occur after one of the following:

a) after the last SPI command information unit is successfully received by a target;
b) after a SPI data information unit is successfully received by or transmitted from a target;
c) after a SPI status information unit is successfully transmitted from a target;
d) after a SPI L_Q information unit if the SPI L_Q information unit DATA LENGTH field is zero; or
e) during a QAS phase.

10.5 Arbitration

10.5.1 Arbitration and QAS overview

Arbitration allows one SCSI device to gain control of the SCSI bus to allow that SCSI device to initiate or resume a task.

There are two methods that a SCSI device may use to arbitrate for the SCSI bus: normal arbitration and QAS. Normal arbitration is mandatory and requires the detection of a BUS FREE phase on the SCSI bus before starting. QAS is optional and, when enabled (see 16.3.10), requires the initiation and detection of a QAS REQUEST message (see 16.3.11) before starting.

SCSI devices with arbitration fairness enabled shall maintain a fairness register which records the SCSI IDs of devices that need a chance to arbitrate (see Annex B). Fairness in normal arbitration is enabled in targets by the Disconnect-Reconnect mode page (see 18.1.2). Fairness is always enabled in QAS.

10.5.2 NORMAL ARBITRATION phase

The procedure for a SCSI device to obtain control of the SCSI bus is as follows:

1) The SCSI device shall first wait for the BUS FREE phase to occur. The BUS FREE phase is detected whenever both the BSY and SEL signals are simultaneously and continuously false for a minimum of one bus settle delay.

NOTE 29 - This bus settle delay is necessary because a transmission line phenomenon known as a wired-OR glitch may cause the BSY signal to briefly appear false, even though it is being driven true.

2) The SCSI device shall wait a minimum of one bus free delay after detection of the BUS FREE phase (i.e. after the BSY and SEL signals are both false for one bus settle delay) before driving any signal.
3) Following the bus free delay in step (2), the SCSI device may arbitrate for the SCSI bus by asserting both the BSY signal and its own SCSI ID. However the SCSI device shall not arbitrate

(i.e., assert the BSY signal and its SCSI ID) during this NORMAL ARBITRATION phase if more than one bus set delay has passed since the BUS FREE phase was last observed. If arbitration fairness is enabled, the SCSI device shall not arbitrate until its fairness register is cleared (see Annex B).

NOTE 30 - There is no maximum delay before asserting the BSY signal and the SCSI ID following the bus free delay in step (2) as long as the bus remains in the BUS FREE phase. However, SCSI devices that delay longer than one bus settle delay plus one bus set delay from the time when the BSY and SEL signals first become false may fail to participate in arbitration when competing with faster SCSI devices and may not be ensured fair arbitration by the arbitration fairness algorithm.

4) After waiting at least an arbitration delay (measured from its assertion of the BSY signal) the SCSI device shall examine the DATA BUS.
   1) If no higher priority SCSI ID bit is true on the DATA BUS, then the SCSI device has won the arbitration and it shall assert the SEL signal.
   2) If a higher priority SCSI ID bit is true on the DATA BUS (see table 28 for the SCSI ID arbitration priorities), then the SCSI device has lost the arbitration and the SCSI device shall release the BSY signal and the SCSI ID after the SEL signal becomes true, within one bus clear delay after the SEL signal becomes true. Any losing SCSI devices may return to step (1).

NOTE 31 - Step (4) above requires any SCSI device that begins NORMAL ARBITRATION phase to complete the NORMAL ARBITRATION phase to the point of SEL being asserted if it begins the NORMAL ARBITRATION phase as stated in step (3). This precludes the possibility of the bus being hung.

5) After the bus free delay in step (2), SCSI devices with arbitration fairness enabled that are not arbitrating shall wait one bus set delay and start sampling the DATA BUS to determine the SCSI devices that attempted arbitration, the SCSI device that won, and the SCSI devices that lost. This sampling shall continue for an arbitration delay after the bus free delay in step (2). Each SCSI device shall update its fairness register with all lower-priority device IDs that lost arbitration.

NOTE 32 - For ease of implementation, this sampling may begin when BSY is true following BUS FREE and end when SEL is true.

6) The SCSI device that wins arbitration shall wait at least one bus clear delay plus one bus settle delay after asserting the SEL signal before changing any signals.

The SCSI ID bit is a single bit on the DATA BUS that corresponds to the SCSI device's unique SCSI address. All other DATA BUS bits shall be released by the SCSI device. During the NORMAL ARBITRATION phase, DB(P_CRCA), and DB(P1) (if present) may be released or asserted, but shall not be actively driven false.

10.5.3 QAS protocol

QAS allows a target that has information unit transfers enabled (see 16.3.10) and QAS enabled (see 16.3.10) that is currently connected to an initiator that has information unit transfers enabled and QAS enabled to transfer control of the bus to another SCSI device that has information unit transfers enabled and QAS enabled without an intervening BUS FREE phase. SCSI devices that support QAS shall report that capability in the INQUIRY command.

In order to enable QAS, an initiator that supports QAS shall negotiate, using the PPR message, the use of the QAS phase with each target that has indicated support of QAS. Any time the data transfer agreement is in an indeterminate state (see 16.3.10) the initiator shall renegotiate to enable QAS. SCSI devices that support QAS shall implement the fairness algorithm (see Annex B) during all QAS arbitrations. SCSI devices shall negotiate the use of QAS with a particular SCSI device before using QAS to select or reselect that SCSI device. Also, targets shall have negotiated the use of QAS with a particular initiator before using QAS REQUEST message to do a physical disconnect from that initiator, and initiators shall have negotiated the use of QAS with a particular target before accepting a QAS REQUEST message from that target. If an initiator receives a QAS REQUEST message from a target that has not negotiated the use

of QAS, then the initiator shall create an attention condition for the QAS REQUEST message, and shall report MESSAGE REJECT on the following MESSAGE OUT phase.

In an environment where some SCSI devices have QAS enabled and other SCSI devices do not, it is possible for the SCSI devices that have QAS enabled to prevent SCSI devices that do not have QAS enabled from arbitrating for the bus. This occurs when SCSI devices that have QAS enabled never go to a BUS FREE phase.

A QAS initiator may interrupt a sequence of QAS cycles to force a normal arbitration with the following procedure:

1) Perform a QAS arbitration;
2) on winning QAS arbitration, continue driving the initiator's ID on the DATA BUS instead of asserting SEL to enter selection phase;
3) wait until the target transitions to BUS FREE (this occurs after two QAS arbitration delays);
4) after detecting BSY false, release the DATA BUS; and
5) after one bus settle delay from when the target drove BSY false, the bus is in BUS FREE phase. The initiator may then arbitrate using normal arbitration and perform a selection if it wins.

10.5.4 QAS phase

The procedure for a target to indicate it wants to release the bus is as follows:

1) The target shall change to a MESSAGE IN phase and issue a single QAS REQUEST (55h) message.
2) After detection of the REQ signal being true for the QAS REQUEST message, the initiator shall assert ACK for at least a QAS non-DATA phase REQ(ACK) period.
3) After detection of the ACK signal being true the target shall negate REQ.
4) After detection of the REQ signal being false, the initiator shall wait at least a QAS non-DATA phase REQ(ACK) period then negate ACK.
5) After detection of the ACK signal being false and if the initiator did not create an attention condition, the target shall release all SCSI signals except the BSY, MSG, C/D, I/O, and REQ signals. Then the target shall negate the MSG, C/D, and I/O signals within two system deskew delays. The target shall wait two system deskew delays after negating the C/D, I/O, and MSG signals before releasing the REQ signal.
6) ~~After the initiator negates the ACK signal for the QAS REQUEST message and if~~ If the initiator did not create an attention condition then the initiator shall release all SCSI signals within two system deskew delays after detecting MSG, C/D, and I/O signals false.
7) ~~After detection of the last ACK signal being true the target shall negate REQ. After detection of the last ACK signal being false, if there is no attention condition, the target shall release all SCSI signals except the BSY, MSG, C/D, I/O, and REQ signals. Then the target shall negate the MSG, C/D, and I/O signals within two system deskew delays. The target shall wait two system deskew delays after negating the C/D, I/O, and MSG signals before releasing the REQ signal.~~
8) If the target detects the SEL signal being true, the target shall release the BSY, MSG, C/D, and I/O signals within one QAS release delay.
9) After waiting at least one QAS arbitration delay from releasing the SCSI signals in step (~~7)~~5), if there are no SCSI ID bits true the target shall transition to the BUS FREE phase.
10) After waiting at least one QAS arbitration delay from releasing the SCSI signals in step (~~7)~~5), if there are any SCSI ID bits true the target shall wait at least a second QAS arbitration delay. If the SEL signal is not true by the end of the second QAS arbitration delay the target shall transition to the BUS FREE phase.

NOTE 33 - The release of MSG, C/D, and I/O may cause release glitches; Step (10) above ensures these glitches occur at a time when no connection is established on the bus so that they do not interfere with proper operation.

The procedure for a SCSI device to obtain control of the SCSI bus is as follows:

1) The SCSI device shall first wait for MESSAGE IN phase to occur with a single QAS REQUEST (55h) message. When the SCSI device detects the ACK signal being false for the QAS REQUEST message and the attention condition is cleared it shall begin the QAS phase.
2) The SCSI device shall wait a minimum of two deskew delays after detection of the MSG, C/D, and I/O signals being false before driving any signal.
3) Following the delay in step (2), the SCSI device may arbitrate for the SCSI bus by asserting its own SCSI ID within one QAS assertion delay from detection of the MSG, C/D, and I/O signals being false. If arbitration fairness is enabled, the SCSI device shall not arbitrate until its fairness register is cleared.
4) After waiting at least one QAS arbitration delay (measured from the detection of the MSG, C/D, and I/O signals being negated) the SCSI device shall examine the DATA BUS.
    1) If no higher priority SCSI ID bit is true on the DATA BUS and the fairness algorithm allowed the SCSI device to participate, then the SCSI device has won the arbitration and it shall assert the SEL signal.
    2) If a higher priority SCSI ID bit is true on the DATA BUS (see table 28 for the SCSI ID arbitration priorities) or the fairness algorithm prevented the SCSI device from participating in QAS arbitration, then the SCSI device has lost the arbitration.
    3) Any SCSI device other than the winner has lost the arbitration and shall release its SCSI ID bit after two deskew delays and within one QAS release delay after detection of the SEL signal being true. A SCSI device that loses arbitration may return to step (a).
5) The SCSI device that wins arbitration shall wait at least one QAS arbitration delay after asserting the SEL signal before changing any signals.
6) After the QAS arbitration delay in step (4), SCSI devices with arbitration fairness enabled that are not arbitrating shall start sampling the DATA BUS to determine the SCSI devices that are attempting arbitration, the SCSI device that won, and the SCSI devices that lost. This sampling shall continue for one bus settle delay plus two system deskew delays. The SCSI devices shall update their fairness register with all device IDs that lost arbitration.

The SCSI ID bit is a single bit on the DATA BUS that corresponds to the SCSI device's unique SCSI address. All other DATA BUS bits shall be released by the SCSI device. The DB(P_CRCA) and DB(P1) are not valid during the QAS phase. During the QAS phase, DB(P_CRCA), and DB(P1) may be released or asserted, but shall not be actively driven false.

10.6 SELECTION phase

10.6.1 Selection Overview

The SELECTION phase allows an initiator to select a target for the purpose of initiating some target function (e.g., READ or WRITE command). During the SELECTION phase the I/O signal is negated to distinguish this phase from the RESELECTION phase.

The SCSI device that won a normal arbitration has both the BSY and SEL signals asserted and has delayed at least one bus clear delay plus one bus settle delay before ending the NORMAL ARBITRATION phase.

The SCSI device that won QAS has the SEL signal asserted and has delayed at least one QAS arbitration delay before ending the QAS phase.

The SCSI device that won the arbitration identifies itself as an initiator by not asserting the I/O signal.

10.6.2 Selection using attention condition

10.6.2.1 Information unit transfers disabled

The initiator shall set the DATA BUS to a value that is the OR of its SCSI ID bit, the target's SCSI ID bit, and the appropriate parity bit(s) (i.e., DB(P_CRCA), and/or DB(P1)). If information unit transfers are

disabled (see 16.3.10) the initiator shall create an attention condition (indicating that a MESSAGE OUT phase is to follow the SELECTION phase).

If the arbitration was a normal arbitration then the initiator shall wait at least two system deskew delays and release the BSY signal. The initiator shall then wait at least one bus settle delay before attempting to detect an assertion of the BSY signal from the target.

The target shall detect it is selected when the SEL signal and its SCSI ID bit are true and the BSY and I/O signals are false for at least one bus settle delay. The selected target may examine the DATA BUS in order to determine the SCSI ID of the selecting initiator. The selected target shall then assert the BSY signal within one selection abort time of its most recent detection of being selected; this is required for correct operation of the selection time-out procedure.

The target shall not respond to a selection if bad parity is detected (see 11.2). Also, if more or less than two SCSI ID bits are on the DATA BUS, the target shall not respond to selection.

No less than two system deskew delays after the initiator detects the BSY signal is true, it shall release the SEL signal and may change the DATA BUS. The target shall wait until the SEL signal is false before asserting the REQ signal to enter an information transfer phase.

If information unit transfers are disabled (see 16.3.10) for the connecting initiator the target shall follow the phase sequences defined in 13.2.

10.6.2.2 Information unit transfers enabled

If information unit transfers are enabled (see 16.3.10) for the connecting initiator the target shall follow the phase sequences defined in 13.5.

If information unit transfers are enabled for the connecting initiator the target shall proceed to a MESSAGE OUT phase. On detecting the MESSAGE OUT phase the initiator shall begin a PPR negotiation (see 16.3.10). On completion of the PPR negotiation the target shall proceed to a BUS FREE phase. If the first message received by the target during the MESSAGE OUT phase is not a task management message or a PPR message the target shall change to a MESSAGE IN phase and issue a MESSAGE REJECT message followed by a WDTR message with TRANSFER WIDTH EXPONENT field set to 00h. If the target does not support the WDTR message it shall follow the MESSAGE REJECT message with a SDTR message with the REQ/ACK OFFSET field set to 00h.

10.6.2.3 Selection using attention condition time-out procedure

Two optional selection time-out procedures are specified for clearing the SCSI bus if the initiator waits a minimum of one selection time-out delay and there has been no BSY signal response from the target:

a) Optionally, the initiator shall assert the RST signal (see 12.3);
b) Optionally, the initiator shall continue asserting the SEL and ATN signals and shall release DATA BUS, DB(P_CRCA), and/or DB(P1). If the initiator has not detected the BSY signal to be true after at least one selection abort time plus two system deskew delays, the initiator shall release the SEL and ATN signals allowing the SCSI bus to go to the BUS FREE phase. SCSI devices shall ensure that when responding to selection that the selection was still valid within one selection abort time of their assertion of the BSY signal. Failure to comply with this requirement could result in an improper selection (two targets connected to the same initiator, wrong target connected to an initiator, or a target connected to no initiator).

10.6.3 Selection without using attention condition

10.6.3.1 Information unit transfers disabled or enabled

The initiator shall set the DATA BUS to a value that is the OR of its SCSI ID bit, the target's SCSI ID bit,

and the appropriate parity bit(s) (i.e., DB(P_CRCA), and/or DB(P1)) and it shall clear the attention condition (indicating that a INFORMATION UNIT OUT phase is to follow the SELECTION phase).

If the arbitration was a normal arbitration then the initiator shall wait at least two system deskew delays and release the BSY signal. The initiator shall then wait at least one bus settle delay before attempting to detect an assertion of the BSY signal from the target.

If QAS was used for arbitration then the initiator shall wait at least one bus settle delay before attempting to detect an assertion of the BSY signal from the target.

The target shall detect it is selected when the SEL signal and its SCSI ID bit are true and the BSY and I/O signals are false for at least one bus settle delay. The selected target may examine the DATA BUS in order to determine the SCSI ID of the selecting initiator. The selected target shall then assert the BSY signal within one selection abort time of its most recent detection of being selected; this is required for correct operation of the selection time-out procedure.

The target shall not respond to a selection if bad parity is detected (see 11.2). Also, if more or less than two SCSI ID bits are on the DATA BUS, the target shall not respond to selection.

No less than two system deskew delays after the initiator detects the BSY signal is true, it shall release the SEL signal and may change the DATA BUS. The target shall wait until the SEL signal is false before asserting the REQ signal to enter an information transfer phase.

If information unit transfers are enabled (see 16.3.10) for the connecting initiator the target shall follow the phase sequences defined in 13.4.

If information unit transfers are disabled (see 16.3.10) for the connecting initiator the target shall follow the phase sequences defined in 13.3.

If an initiator, when selecting without using an attention condition, detects an unexpected COMMAND phase it should invalidate all prior negotiations with the selected target. In this case, the initiator shall create an attention condition and on the corresponding MESSAGE OUT phase shall issue an ABORT TASK message. On the next selection of the target that received the ABORT TASK message the initiator should do a selection using the attention condition and negotiate to enable information unit transfers.

10.6.3.2 Selection without using attention condition time-out procedure

Two optional selection time-out procedures are specified for clearing the SCSI bus if the initiator waits a minimum of one selection time-out delay and there has been no BSY signal response from the target:

a) Optionally, the initiator shall assert the RST signal (see 12.3);
b) Optionally, the initiator shall continue asserting the SEL signal and shall release the DATA BUS, DB(P_CRCA), or DB(P1). If the initiator has not detected the BSY signal to be true after at least one selection abort time plus two system deskew delays, the initiator shall release the SEL signal allowing the SCSI bus to go to the BUS FREE phase. SCSI devices shall ensure that when responding to selection that the selection was still valid within one selection abort time of their assertion of the BSY signal. Failure to comply with this requirement could result in an improper selection (two targets connected to the same initiator, wrong target connected to an initiator, or a target connected to no initiator).

10.7 RESELECTION phase

10.7.1 RESELECTION phase overview

The RESELECTION phase allows a target to physically reconnect to an initiator for the purpose of continuing some operation that was previously started by the initiator but was suspended by the target,

(i.e., the target physically disconnected by allowing a BUS FREE phase to occur or issued a QAS REQUEST message before the operation was complete). During the RESELECTION phase the I/O signal is asserted to distinguish this phase from the SELECTION phase.

10.7.2 Physical reconnection

The SCSI device that won a normal arbitration has both the BSY and SEL signals asserted and has delayed at least one bus clear delay plus one bus settle delay before ending the NORMAL ARBITRATION phase.

The SCSI device that won a QAS has the SEL signal asserted and has delayed at least a QAS arbitration delay before ending the QAS phase.

The SCSI device that won the arbitration identifies itself as a target by asserting the I/O signal.

The winning SCSI device shall also set the DATA BUS to a value that is the logical OR of its SCSI ID bit and the initiator's SCSI ID bit and the appropriate parity bit(s) (i.e., DB(P_CRCA), and/or DB(P1)).

If the arbitration was a normal arbitration then target shall wait at least two system deskew delays and release the BSY signal. The target shall then wait at least one bus settle delay before attempting to detect an assertion of the BSY signal by the initiator.

If QAS was used for arbitration then the target shall wait at least one bus settle delay before attempting to detect an assertion of the BSY signal from the initiator.

The initiator shall be physically reconnected when the SEL and I/O signals and its SCSI ID bit are true and the BSY signal is false for at least one bus settle delay. The physically reconnected initiator may examine the DATA BUS in order to determine the SCSI ID of the physically reconnected target. The physically reconnected initiator shall then assert the BSY signal within one selection abort time of its most recent detection of being physically reconnected; this is required for correct operation of the time-out procedure.

The initiator shall not respond to a physical reconnection if bad parity is detected (see 11.2). Also, if more than or less than two SCSI ID bits are on the DATA BUS, the initiator shall not respond to a physical reconnection.

After the target detects the assertion of the BSY signal, it shall also assert the BSY signal and wait at least two system deskew delays and then release the SEL signal. The target may then change the I/O signal and the DATA BUS. After the physically reconnected initiator detects the SEL signal is false, it shall release the BSY signal. The target shall continue asserting the BSY signal until it relinquishes the SCSI bus.

> NOTE 34 - When the target is asserting the BSY signal, a transmission line phenomenon known as a wired-OR glitch may cause the BSY signal to appear false for up to a round-trip propagation delay following the release of the BSY signal by the initiator. This is the reason why the BUS FREE phase is recognized only after both the BSY and SEL signals are continuously false for a minimum of one bus settle delay. For more information on glitches see 7.2.3 and 7.3.4.1.

10.7.3 Physical reconnection time-out procedure

Two optional physical reconnection time-out procedures are specified for clearing the SCSI bus during a RESELECTION phase if the target waits a minimum of one selection time-out delay and there has been no BSY signal response from the initiator:

a) Optionally, the target shall assert the RST signal (see 12.3);
b) Optionally, the target shall continue asserting the SEL and I/O signals and shall release all DATA BUS, DB(P_CRCA), and/or DB(P1) signals. If the target has not detected the BSY signal to be true after at least one selection abort time plus two system deskew delays, the target shall release the SEL and I/O signals allowing the SCSI bus to go to the BUS FREE phase. SCSI devices shall

ensure that the physical reconnection was still valid within one selection abort time of their assertion of the BSY signal. Failure to comply with this requirement could result in an improper physical reconnection (two initiators connected to the same target or the wrong initiator connected to a target).

10.8 Information transfer phases

10.8.1 Information transfer phases overview

The COMMAND, DATA, STATUS, and MESSAGE phases are all grouped together as the information transfer phases because they are all used to transfer data or control information via the DATA BUS. The actual content of the information is beyond the scope of this subclause.

The C/D, I/O, and MSG signals are used to distinguish between the different information transfer phases (see table 34). The target drives these three signals and therefore controls all changes from one phase to another. The initiator requests a MESSAGE OUT phase creating an attention condition. The target causes the BUS FREE phase by releasing the MSG, C/D, I/O, and BSY signals.

The information transfer phases use one or more REQ or ACK handshakes to control the information transfer. Each REQ/ACK handshake allows the transfer of 8- or 16-bits of information depending on the negotiated data transfer width (see 16.3.16). During the information transfer phases the BSY signal shall remain true and the SEL signal shall remain false. Additionally, during the information transfer phases, the target shall continuously envelope the REQ or ACK handshake(s) with the C/D, I/O, and MSG signals in such a manner that these control signals are valid for one bus settle delay before the assertion of the REQ signal of the first handshake and remain valid until after the negation of the ACK signal at the end of the handshake of the last transfer of the phase.

The target shall not transition into an information transfer phase unless the REQ and ACK signals are negated. The target shall not transition from an information transfer phase into another information transfer phase unless the REQ/ACK signals are negated.

NOTE 35 - After the negation of the ACK signal of the last transfer of the phase, the target may prepare for a new phase by asserting or negating the C/D, I/O, and MSG signals. These signals may be changed together or individually. They may be changed in any order and may be changed more than once. It is desirable that each line change only once. A new phase does not begin until the REQ signal is asserted for the first byte of the new phase.

NOTE 36 - A phase is defined as ending when the C/D, I/O, or MSG signals change after the negation of the ACK signal. The time between the end of a phase and the assertion of the REQ signal beginning a new phase is undefined.

There are three methods of transferring data using information transfers:

a) asynchronous transfers (see 10.8.2);
b) synchronous transfers (see 10.8.3); and
c) paced transfers (see 10.8.4).

Synchronous transfers shall only be used for negotiated transfer rates less than or equal to fast-80.

Paced transfers shall only be used for a negotiated transfer rate of fast-160.

Table 34 - Information transfer phases

Signal			Phase	Direction of transfer	Comment	
C/D	MSG	I/O				
0	0	0	ST DATA OUT	Initiator to target	ST DATA phase	DATA phase
0	0	1	ST DATA IN	Initiator from target		
0	1	0	DT DATA OUT	Initiator to target	DT DATA phase	
0	1	1	DT DATA IN	Initiator from target		
1	0	0	COMMAND	Initiator to target		
1	0	1	STATUS	Initiator from target		
1	1	0	MESSAGE OUT	Initiator to target	MESSAGE phase	
1	1	1	MESSAGE IN	Initiator from target		
Key: 0 = False; 1 = True						

10.8.2 Asynchronous transfer

The target shall control the direction of information transfer by means of the I/O signal. When the I/O signal is true, information shall be transferred from the target to the initiator. When the I/O signal is false, information shall be transferred from the initiator to the target.

If the I/O signal is true (transfer to the initiator), the target shall first drive the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals to their values, delay at least one system deskew delay plus one cable skew, then assert the REQ signal. The DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals shall remain valid until the ACK signal is true at the target. The initiator shall read the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals after the REQ signal is true, then indicate its acceptance of the data by asserting the ACK signal. When the ACK signal becomes true at the target, the target may change or release the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals and shall negate the REQ signal. After the REQ signal is false the initiator shall then negate the ACK signal. After the ACK signal is false the target may continue the transfer by driving the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals and asserting the REQ signal, as described above.

If the I/O signal is false (transfer to the target) the target shall request information by asserting the REQ signal. The initiator shall drive the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals to their values, delay at least one system deskew delay plus one cable skew and assert the ACK signal. The initiator shall continue to drive the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals until the REQ signal is false. When the ACK signal becomes true at the target, the target shall read the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals then negate the REQ signal. When the REQ signal becomes false at the initiator, the initiator may change or release the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals and shall negate the ACK signal. After the ACK signal is false the target may continue the transfer by asserting the REQ signal, as described above.

10.8.3 Synchronous transfer

10.8.3.1 Synchronous transfer overview

Synchronous transfer is optional and is only used in DATA phases. It shall be used in a DATA phase if a synchronous transfer agreement has been established (see 16.3.14 or 16.3.10). The agreement specifies the REQ/ACK offset and the minimum transfer period.

When synchronous transfers are being used data may be transferred using ST data transfers or, optionally, DT data transfers. DT data transfers shall only be used on 16 bit wide buses that transmit and receive data using LVD transceivers.

Implementors shall not use the following subclauses for timing requirements. For timing requirements see 9.2.

10.8.3.2 ST synchronous transfer

When a ST data transfer agreement has been established the target shall only use the ST DATA IN phase and ST DATA OUT phase for data transfers.

The REQ/ACK offset specifies the maximum number of REQ assertions that shall be sent by the target in advance of the number of ACK assertions received from the initiator. If the number of REQ assertions exceeds the number of ACK assertions by the REQ/ACK offset, the target shall not assert the REQ signal until after the next ACK assertion is received. For successful completion of the ST DATA phase the number of ACK and REQ assertions shall be equal.

For the timing requirements of the negotiated transfer period see 9.2.

If the I/O signal is true (transfer to the initiator), the target shall first drive the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals to their values, wait at least one transmit setup time, then assert the REQ signal. The DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals shall be held valid for a minimum of one transmit hold time after the assertion of the REQ signal. The target shall assert the REQ signal for a minimum of one transmit assertion period. The target may then negate the REQ signal and change or release the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals. The initiator shall read the value on the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals within one receive hold time of the transition of the REQ signal to true. The initiator shall then respond with an ACK assertion.

If the I/O signal is false (transfer to the target), the initiator, after detecting a REQ assertion, shall first drive the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals to their values, delay at least one transmit setup time, then assert the ACK signal. The initiator shall hold the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals valid for at least one transmit hold time after the assertion of the ACK signal. The initiator shall assert the ACK signal for a minimum of one transmit assertion period. The initiator may then negate the ACK signal and may change or release the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals. The target shall read the value of the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals within one receive hold time of the transition of the ACK signal to true.

10.8.3.3 DT synchronous transfer

10.8.3.3.1 DT synchronous transfer overview

When a DT data transfer agreement has been established the target shall only use the DT DATA IN phase and DT DATA OUT phase for data transfers.

During DT data transfers data shall be clocked on both the assertion and negation of the REQ and ACK signal lines. References to REQ or ACK transitions in this subclause refer to either an assertion or a negation of the REQ or ACK signal.

The REQ/ACK offset specifies the maximum number of REQ transitions that shall be sent by the target in advance of the number of ACK transitions received from the initiator. If the number of REQ transitions exceeds the number of ACK transitions by the REQ/ACK offset, the target shall not transition the REQ signal until after the next ACK transition is received. For successful completion of the DT DATA phase the number of ACK and REQ transitions shall be equal and both REQ and ACK shall be negated.

For the timing requirements of the negotiated transfer period see 9.2.

10.8.3.3.2 Information unit transfer

When information unit transfers are enabled (see 16.3.10);

a) Information units shall be transferred on the DT DATA OUT phase and the DT DATA IN phase, and
b) the information units' embedded iuCRC shall be used to detect information unit data errors.

If the I/O signal is true (transfer to the initiator), to transfer SPI information units the target:

1) Shall drive the DB(15-0) signals to their values;
2) shall wait at least one transmit setup time from DB(15-0) being driven with valid data;
3) shall transition the REQ signal;
4) shall hold the DB(15-0) signals valid for a minimum of one transmit hold time;
5) may change or release the DB(15-0) signals; and
6) shall not change the REQ signal for a minimum of one transmit assertion period.

If the I/O signal is true (transfer to the initiator), to receive SPI information units the initiator shall:

1) Read the value on the DB(15-0) signals within one receive hold time of the transition of the REQ signal; and
2) respond with an ACK transition.

If the I/O signal is false (transfer to the target), to transfer SPI information units the initiator:

1) Shall wait until detecting a REQ transition;
2) shall drive the DB(15-0) signals to their values;
3) shall delay at least one transmit setup time;
4) shall transition the ACK signal;
5) shall hold the DB(15-0) signals valid for at least one transmit hold time;
6) shall not change the ACK signal for a minimum of one transmit assertion period; and
7) may then change or release the DB(15-0) signals.

If the I/O signal is false (transfer to the target), to receive SPI information units the target:

1) Shall read the value of the DB(15-0) signals within one receive hold time of the transition of the ACK~~.~~; and
~~2) shall not transition the REQ signal for the current SPI information unit until the initiator has responded with all ACK transitions for the previous SPI information unit.~~
3) the REQ/ACK offset shall be zero at the beginning and end of all SPI information units except for consecutive SPI data stream information units that.

As a result of a SPI information unit always being an even number of transfers, the REQ and ACK signals are negated both before and after the transmission of the SPI information unit.

10.8.3.3.3 DT DATA IN phase information unit transfer exception condition handling

The initiator shall not negate the ACK for the last byte of the last iuCRC in an information unit until the entire information unit has been verified and any required attention condition has been established.

If the nexus has been fully identified (i.e., an I_T_L_Q nexus has been established) and the initiator detects an iuCRC error in any information unit, other than a SPI status information unit, it receives while in the DT DATA IN phase the initiator shall create an attention condition on or before the last iuCRC, within the failed information unit, is acknowledged. When the target switches to a MESSAGE OUT phase the initiator should send an INITIATOR DETECTED ERROR message (see 16.3.5) to the target. This message notifies the target that data in the information unit was invalid.

If an initiator detects an iuCRC error in a SPI status information unit the initiator shall create an attention condition on or before the last iuCRC of the information unit is acknowledged. If the target detects an attention condition it shall switch to a MESSAGE OUT phase and the initiator shall send an INITIATOR DETECTED ERROR message (see 16.3.5) or an ABORT TASK message to the target. These messages notify the target that the SPI status information unit was invalid.

If the information unit that failed was not a SPI status information unit and the message received from the initiator was an INITIATOR DETECTED ERROR message then the target shall send a SPI L_Q/SPI status information unit pair to the initiator with a CHECK CONDITION status and a sense key set to ABORTED COMMAND and an additional sense code set to INITIATOR DETECTED ERROR MESSAGE RECEIVED for the task associated with the received INITIATOR DETECTED ERROR message.

If the information unit that failed was a SPI status information unit and the message received was an INITIATOR DETECTED ERROR message then the target shall retry transferring the SPI L_Q/SPI status information unit pair to the initiator with the original status information.

If the information unit that failed was a SPI status information unit and the message received was an ABORT TASK message then the target shall cause a bus free by generating a BUS FREE phase.

If the initiator is receiving a SPI L_Q information unit and the initiator detects an iuCRC error (i.e., the nexus identification fails) while in the DT DATA IN phase the initiator shall create the attention condition on or before the last iuCRC. When the target switches to a MESSAGE OUT phase the initiator shall send an INITIATOR DETECTED ERROR message (see 16.3.5) to the target. This message notifies the target that the nexus identification failed. The target shall then cause a bus free by generating a BUS FREE phase. The target shall retry the task associated with the failed SPI L_Q information unit.

If the initiator receives a SPI L_Q information unit with a type code that is not defined in table 41 that initiator shall create an attention condition after negating the ACK for the last byte of the iuCRC in the SPI L_Q information unit and before negating the ACK for the last byte of the last iuCRC in the information unit that follows the SPI L_Q information unit. When the target switches to a MESSAGE OUT phase the initiator shall send an ABORT TASK message (see 16.5.2) to the target. The target shall cause a bus free by generating a BUS FREE phase.

10.8.3.3.4 DT DATA OUT phase information unit transfer exception condition handling

The target shall only respond to an iuCRC error after all the data in an information unit has been received.

If the nexus has been fully identified (i.e., an I_T_L_Q nexus has been established) and the target detects an iuCRC error in any SPI information unit it receives while in the DT DATA OUT phase the target shall, before receiving another SPI L_Q information unit, switch to a DT DATA IN phase and send a SPI L_Q/SPI status information unit pair to the initiator with a CHECK CONDITION status and a sense key set to ABORTED COMMAND and the additional sense code set to iuCRC ERROR DETECTED for the task associated with the iuCRC error.

If the target detects an iuCRC error on an iuCRC interval that is not at the end of a SPI information unit the target shall not respond to the error until all the bytes of the SPI information unit in which the error occurred have been transferred. The target may discard the transmitted information.

If the target is receiving a SPI L_Q information unit and the target detects an iuCRC error (i.e., the nexus identification fails) the target shall cause an unexpected bus free by generating a BUS FREE phase (see

10.3).

If a target receives a SPI L_Q information unit with a type code that is not defined in table 41 that target shall transfer all the bytes indicated by the data length and iuCRC interval and shall discard the transmitted information. After transferring all the bytes the target shall change to a DT DATA IN phase and transmit a SPI status information unit with a RSPVALID bit of one and the packetized failure code set to INVALID TYPE CODE RECEIVED IN SPI L_Q INFORMATION UNIT.

10.8.3.3.5 Data Group data field transfer

When the target is transferring consecutive data groups, it shall not transition the REQ signal while the P_CRCA signal is asserted for the current data group until the initiator has acknowledged the entire previous data group.

NOTE 37 - The requirement above ensures the initiator is not required to maintain more than one simultaneous pCRC calculation in different data groups.

If the I/O signal is true (transfer to the initiator), to transfer the data field the target:

1) Shall drive the DB(15-0) signals to their values and shall negate the P_CRCA signal;
2) shall wait at least the longer of one pCRC transmit setup time from the negation of P_CRCA or one transmit setup time from DB(15-0) being driven with valid data;
3) shall transition the REQ signal;
4) shall hold the DB(15-0) signals valid for a minimum of one transmit hold time and shall hold the P_CRCA signal for a minimum of one pCRC transmit hold time;
5) may change or release the DB(15-0) and P_CRCA signals; and
6) shall not change the REQ signal for at least one transmit assertion period if asserted or one transmit negation period if negated.

If the I/O signal is true (transfer to the initiator), to receive the data field the initiator shall:

1) Read the value on the DB(15-0) signals within one receive hold time of the transition of the REQ signal;
2) Read the value of the P_CRCA signal within one pCRC receive hold time of the transition of the REQ signal; and
3) respond with an ACK transition.

If the I/O signal is false (transfer to the target), to transfer the data field the initiator:

1) Shall wait until detecting a REQ transition with P_CRCA negated;
2) shall drive the DB(15-0) signals to their values;
3) shall delay at least one transmit setup time;
4) shall transition the ACK signal;
5) shall hold the DB(15-0) signals valid for at least one transmit hold time;
6) may then change or release the DB(15-0) signals; and
7) shall not change the ACK signal for at least one transmit assertion period if asserted or one transmit negation period if negated.

If the I/O signal is false (transfer to the target), to receive the data field the target:

1) Shall read the value of the DB(15-0) signals within one receive hold time of the transition of the ACK.

10.8.3.3.6 Data Group Pad field and pCRC field transfer to initiator

The target detects a pad field is required if the I/O signal is true (transfer to the initiator), the target has completed the data field transfer of the current data group, and REQ signal is asserted. In this case the

target shall:

1) Wait at least one pCRC transmit hold time since the last REQ assertion to assert P_CRCA;
2) wait at least one transmit hold time since the last REQ assertion to assert the DB(15-0) signals to their pad values;
3) wait at least the longer of one pCRC transmit setup time from the assertion of P_CRCA or one transmit setup time from DB(15-0) being driven with valid pad data;
4) wait until the initiator has responded with all ACK transitions for the previous data group;
5) wait at least one transmit REQ assertion period with P_CRCA transitioning since the last REQ assertion;
6) negate the REQ signal;
7) hold the DB(15-0) signals valid for a minimum of one transmit hold time and hold the REQ signal negated for a minimum of one transmit negation period;
8) drive the DB(15-0) signals to their pCRC values;
9) wait at least one transmit setup time;
10) assert the REQ signal;
11) hold the DB(15-0) signals for a minimum of one transmit hold time and hold the REQ signal asserted for a minimum of one transmit assertion period;
12) drive the DB(15-0) signals to their pCRC values;
13) wait at least one transmit setup time;
14) negate the REQ signal;
15) hold the DB(15-0) signals for a minimum of one transmit hold time and hold the P_CRCA signal asserted for at least one pCRC transmit hold time; and
16) hold the REQ signal negated for at least one transmit REQ negation period with P_CRCA transitioning since the last REQ negation.

If the target detects no pad field is required (i.e., the REQ signal is negated) the target shall:

1) Wait at least one pCRC transmit hold time since the last REQ negation to assert P_CRCA;
2) wait at least one transmit hold time since the last REQ negation to assert the DB(15-0) signals to their pCRC values;
3) wait at least the longer of one pCRC transmit setup time from the assertion of P_CRCA or one transmit setup time from DB(15-0) being driven with valid pCRC data;
4) wait until the initiator has responded with all ACK transitions for the previous data group;
5) wait at least one transmit REQ negation period with P_CRCA transitioning since the last REQ negation;
6) assert the REQ signal;
7) hold the DB(15-0) signals for a minimum of one transmit hold time and hold the REQ signal asserted for a minimum of one transmit assertion period;
8) drive the DB(15-0) signals to their pCRC values;
9) wait at least one transmit setup time;
10) negate the REQ signal; and
11) hold the DB(15-0) signals for a minimum of one transmit hold time and hold the P_CRCA signal asserted for a minimum of one pCRC transmit hold time.
12) hold the REQ signal negated for at least one transmit REQ negation period with P_CRCA transitioning since the last REQ negation

After either of the above sequences is complete the target has ended a data group.

The initiator shall read the value on the DB(15-0) signals within one receive hold time of the transition of the REQ signal. The initiator shall then respond with an ACK transition.

The initiator shall continue to use the pad bytes, if any, for checking against the computed pCRC for the current data group. Upon receipt of the last byte of the pCRC field, the received pCRC and computed pCRC shall be compared. If they do match (i.e., no pCRC error). then the initiator shall negate the ACK signal.

If received pCRC and computed pCRC do not match (i.e., a pCRC error is detected), or if an improperly formatted data group is transferred, then the initiator shall create an attention condition or before the last transfer of the data group. When the target switches to a MESSAGE OUT phase the initiator should send an INITIATOR DETECTED ERROR message (see 16.3.5) to the target. This message notifies the target that data contained within the data group was invalid.

If the target does not retry transferring the information transfer or it exhausts its retry limit the target shall go into a STATUS phase and send a CHECK CONDITION status with a sense key set to ABORTED COMMAND and an additional sense code set to INITIATOR DETECTED ERROR MESSAGE RECEIVED for the task associated with the received INITIATOR DETECTED ERROR message.

10.8.3.3.7 Data Group Pad field and pCRC field transfer to target

If the I/O signal is false (transfer to the target), the initiator determines the data field transfer is complete by detecting an assertion of the P_CRCA signal. If the REQ signal is asserted (i.e., pad field required) the initiator shall first transfer the two pad bytes, then the four pCRC bytes. If the REQ signal is negated (i.e., no pad field required) the initiator shall transfer the four pCRC bytes.

Pad field data and pCRC field data are transferred using the same negotiated values as the data field data.

The target may continue to send REQs, up to the negotiated offset, for the next data group. The target shall not transition REQ with P_CRCA asserted until the initiator has responded with all ACK transitions for the previous data group.

When the initiator detects an assertion of the P_CRCA signal and the REQ signal is asserted (i.e., pad field required) it shall then:

1) Transfer data bytes for all outstanding REQs received prior to the REQ that had the P_CRCA signal asserted;
2) drive the DB(15-0) signals to their pad values;
3) delay at least one transmit setup time;
4) negate the ACK signal;
5) hold the DB(15-0) signals valid for a minimum of one transmit hold time and hold the ACK signal negated for a minimum of one transmit assertion period;
6) drive the DB(15-0) signals to their pCRC values;
7) delay at least one transmit setup time;
8) assert the ACK signal;
9) hold the DB(15-0) signals valid for a minimum of one transmit hold time and hold the ACK signal asserted for a minimum of one transmit assertion period;
10) drive the DB(15-0) signals to their pCRC values;
11) delay at least one transmit setup time;
12) negate the ACK signal; and
13) hold the DB(15-0) signals valid for a minimum of one transmit hold time and hold the ACK signal negated for a minimum of one transmit assertion period.

When the initiator detects an assertion of the P_CRCA signal and the REQ signal is negated (i.e., no pad field required) it shall then:

1) Transfer data bytes for all outstanding REQs received prior to the REQ that had the P_CRCA signal asserted;
2) drive the DB(15-0) signals to their pCRC values;
3) delay at least one transmit setup time;
4) assert the ACK signal;
5) hold the DB(15-0) signals valid for a minimum of one transmit hold time and hold the ACK signal asserted for a minimum of one transmit assertion period;
6) drive the DB(15-0) signals to their pCRC values;
7) delay at least one transmit setup time;

8) negate the ACK signal; and
9) hold the DB(15-0) signals valid for a minimum of one transmit hold time and hold the ACK signal negated for a minimum of one transmit assertion period.

After either of the above sequences is complete the target has ended a data group.

As a result of a data group always being an even number of transfers, the REQ and ACK signals are negated both before and after the transmission of the data group.

The target shall read the value of the DB(15-0) signals within one receive hold time of the transition of the ACK signal.

The initiator shall use the pad bytes, if any, in the generation of the transmitted pCRC. The target shall then use those pad bytes, if any, for checking against the computed pCRC for the current data group. Upon receipt of the last byte of the pCRC field, the received pCRC and computed pCRC shall be compared.

If received pCRC and computed pCRC do not match (i.e., a pCRC error is detected), or if an improperly formatted data group is transferred, then the associated data group shall be considered invalid.

If the target does not retry transferring the information transfer or it exhausts its retry limit the target shall go into a STATUS phase and send a CHECK CONDITION status with a sense key set to ABORTED COMMAND and an additional sense code set to SCSI PARITY ERROR for the task associated with the pCRC error.

10.8.4 Paced transfer

10.8.4.1 Paced transfer overview

~~Paced transfer is optional and is only used in DT DATA phases. It shall be used in a DT DATA phase if a paced transfer agreement has been established. (see 16.3.14 or 16.3.10). The agreement specifies the REQ/ACK offset and the transfer period.~~

If a paced transfer agreement has been established it shall be used in DT DATA phase and information unit transfers shall be used. The agreement also specifies the REQ/ACK offset and the transfer period (see 16.3.10).

When paced transfers are being used data shall be transferred using DT data transfers on ~~16~~ 16-bit wide buses that transmit and receive data using LVD transceivers.

If driver precompensation is enabled at the SCSI device, that SCSI device shall apply driver precompensation to all the data, parity, REQ, and ACK signals.

During paced DT data transfers, if the phase of the P1 signal indicates data is valid (see 10.8.4.3) on REQ or ACK ~~transitions~~assertions, data shall be clocked by the originating SCSI device by both the assertion and negation of the REQ ~~and~~ or ACK signal lines. The receiving SCSI device shall clock DT data on both the assertion and negation of the REQ or ACK signal line after having been processed by the receiving SCSI device. If the phase of the P1 signal indicates data is invalid on REQ or ACK ~~transitions~~assertions, data shall not be clocked by the originating SCSI device and shall be ignored by the receiving SCSI device. If driver precompensation is enabled at the originating SCSI device, the originating SCSI device shall apply driver precompensation to all the data signals, the P_CRCA signal, the P1 signal, and the REQ, and or ACK signal.

~~If driver precompensation is enabled at the originating SCSI device, the originating SCSI device shall apply driver precompensation to all the data, parity, REQ, and ACK signals.~~

~~References to REQ or ACK transitions in this subclause refer to either an assertion or a negation of the REQ or ACK signal.~~

For paced DT DATA IN phases the REQ/ACK offset specifies the maximum number of data valid state REQ assertions (see 10.8.4.3) that shall be sent by the target in advance of the number of ACK assertions received from the initiator. If the number of data valid state REQ assertions exceeds the number of ACK assertions by the REQ/ACK offset, the target shall change P1 to enable the data invalid state prior to the next assertion of REQ and shall not change P1 to enable a data valid state until after the next ACK assertion is received. For successful completion of a paced DT DATA IN phase the number of data valid state REQ assertions and ACK assertions shall be equal. Each assertion indicates a single 32-bit data transfer.

~~The~~ For paced DT DATA OUT phases the REQ/ACK offset specifies the maximum number of REQ ~~transitions~~ assertions that shall be sent by the target in advance of the number of data valid state ACK ~~transitions~~ assertions received from the initiator. If the number of REQ ~~transitions~~ assertions exceeds the number of data valid state ACK ~~transitions~~ assertions by the REQ/ACK offset, the target shall not ~~transition the~~ assert REQ ~~signal~~ until after the next data valid state ACK ~~transition~~ assertion is received. For successful completion of ~~the~~ a paced DT DATA OUT phase the number of ~~ACK and~~ REQ ~~transitions shall be equal~~ assertions and ~~both REQ and~~ data valid state ACK assertions shall be ~~negated~~equal. Each assertion indicates a single 32-bit data transfer.

Implementors shall not use the following subclauses for timing requirements. For timing requirements see 9.2.

10.8.4.2 Paced transfer training pattern

10.8.4.2.1 Training pattern overview

If the retain training information is disabled a training pattern shall be transferred at the start of the first DT data phase for each data transfer direction of an I_T nexus. The training pattern shall not be transferred again after the start of the first DT data phase for each data transfer direction of an I_T nexus.

If the retain training information is enabled a training pattern shall be transferred at the start of the first DT data phase for each data transfer direction after the retain training information is enabled. The SCSI device shall save training configuration values for each I_T nexus that has negotiated to retain training information. The SCSI device shall use the saved training configuration values for all paced transfers. The target may retrain an I_T nexus if it determines the training configuration values are invalid, without having to renegotiate the retain training information protocol option.

NOTE 38 - The training configuration values are vender specific.

~~A~~ The training pattern ~~shall be transferred at the start of the first DT data~~ for a DATA IN phase ~~for each data transfer direction of an I_T nexus. The training pattern~~ shall ~~not be transferred again after the start of the first DT data phase for each data transfer direction of an I_T nexus. The training pattern for a DATA~~ conform to 10.8.4.2.2. The training pattern for a DATA ~~The training pattern for a DATA~~ OUT phase shall conform to ~~10.8.4.2.3~~10.8.4.2.3. The receiving SCSI device shall use some or all elements of the training pattern to achieve deskewing (see 10.8.4.5). The transmitting SCSI device shall not make an intentional shift in relative timing between the data bus signals and the REQ or ACK signal during the DT data phase.

NOTE 39 - The requirement to not intentionally change relative timing does not include the effects of ISI, noise, or jitter.

The training pattern consists of three sections; A, B, and C. Each section contains a different pattern that may be used to train circuits within a receiver. The B training section is used to deskew the REQ, P1, and DB(15-0,P_CRCA) signals. The use of the A and C training sections is implementation specific and not defined but this standard.

10.8.4.2.2 DT DATA IN phase training pattern

~~The target shall indicate a training pattern is going to occur on a DT DATA IN phase by:~~

~~1) negating the SEL signal a minimum of two system deskew delays before changing the MSG, C/D, and I/O signals, and~~

The target shall indicate a training pattern is going to occur on a DT DATA IN phase by asserting the SEL signal a minimum of two system deskew delays ~~after~~ before asserting the ~~MSG and I/O signals and negating the C/D~~REQ signal.

~~The target shall begin the A section of the training pattern no sooner than two system deskew delays after negating the SEL signal. The target shall transmit the following training pattern:~~

The target shall begin the section A of the training pattern only after all the signal restrictions between information transfers phases listed in 10.13 or the signal restrictions between a RESELECTION phase and a DT DATA IN phase listed in 10.7.2 are met. The target shall transmit the following training pattern:

Start of ~~A section~~section A;
1) disable precompensation;
2) simultaneously assert REQ, P1, and DB(15-0,P_CRCA) signals;
3) wait the equivalent of 32 transfer periods (e.g., 200 ns at fast-160);
4) ~~s~~imultaneously negate REQ, P1, and DB(15-0,P_CRCA) signals;
5) wait the equivalent of 32 transfer periods;
6) set precompensation to negotiated state;
7) negate SEL signal;
8) simultaneously assert and negate REQ, P1, and DB(15-0,P_CRCA) signals at the negotiated ~~rate for 128~~ transfer ~~periods~~period 64 times, (e.g., (2 x 6.25 ns) x 64 = 800 ns at fast-160);

Start of ~~B section~~section B;
1) wait the equivalent of 192 transfer periods from the first assertion of REQ in step 2 of section A (e.g., 1200 ns at fast-160);
2) ~~on~~ keep the ~~128th transfer period negate~~ P1, and DB(15-0,P_CRCA) signals negated while continuing to assert and negate REQ at the negotiated ~~rate~~ transfer period for the equivalent of 8 transfer periods (e.g., 50 ns at fast-160);
3) ~~negate~~ keep the REQ signal negated for the equivalent of 8 additional transfer periods;
4) simultaneously assert and negate P1 and DB(15-0,P_CRCA) signals at twice the negotiated ~~rate for while asserting and negating REQ at the negotiated rate for 48~~ transfer ~~periods~~ period (i.e., simultaneously repeat a 1100b bit pattern 12 times on each signal) while asserting and negating REQ at the negotiated transfer period 24 times (e.g., (2 x 6.25 ns) x 24 = 300 ns at fast-160);

Start of ~~C section~~section C;
~~5) assert and negate REQ at the negotiated rate for 128 transfer periods while repeating a 0000010011111011b bit pattern eight times on P1 and DB(15-0,P_CRCA); and~~
~~6) After ACK negated for greater than 100ns training pattern ends.~~
7) assert and negate REQ at the negotiated transfer period 64 times and at the same time assert and negate P1 at twice the negotiated transfer period while repeating a 0000010011111011b bit pattern eight times on each of the DB(15-0,P_CRCA) signals (e.g., (2 x 6.25 ns) x 64 = 800 ns at fast-160); and

The initiator shall begin the training pattern ~~on detection of~~ if it detects the ~~assertion of the SEL signal with MSG~~ SEL, MSG, and I/O true and C/D ~~false~~false on the first assertion of the REQ signal. The initiator shall transmit the following training pattern:

~~1) assert ACK signal;~~
1) assert ACK signal within 200 ns of the first REQ assertion;
2) disable precompensation;
3) wait the equivalent of 32 transfer periods (e.g., 200 ns at fast-160);
4) negate ACK signal;
5) wait the equivalent of 32 transfer periods (e.g., 200 ns at fast-160); ~~and~~
6) set precompensation to negotiated state; and
7) assert and negate ACK signal at the negotiated ~~rate for 64~~ transfer ~~periods~~period 32 times, (e.g., (2 x 6.25) x 32 = 400 ns at fast-160).

At the completion of the training pattern the target continues asserting and negating the REQ signal at the negotiated ~~rate~~ transfer period (e.g., 6,25 ns transfer period at fast-160) and the P1 signal at twice the negotiated ~~rate~~ transfer period (e.g., 12 ns transfer period at fast-160). When the target is ready to transfer data it shall reverse the phase of P1 (see 10.8.4.3).

10.8.4.2.3 DT DATA OUT phase training pattern

~~The target shall request a training pattern on a DT DATA OUT phase by:~~

~~1) negating the SEL signal a minimum of two system deskew delays before changing the MSG, C/D, and I/O signals, and~~
~~2) asserting the SEL signal a minimum of two system deskew delays after asserting the MSG signal and negating the C/D and I/O signals.~~

~~The target shall begin the training pattern no sooner than two system deskew delays after negating the SEL signal. The target shall transmit the following training pattern:~~

The target shall request a training pattern on a DT DATA OUT phase by asserting the SEL signal a minimum of two system deskew delays before asserting the REQ signal.

The target shall begin the training pattern only after all the signal restrictions between information transfer phases listed in 10.13 or the signal restrictions between a SELECTION phase and a DT DATA OUT phase listed in 10.6.3 are met. The target shall transmit the following training pattern:

1) disable precompensation;
2) simultaneously assert REQ and P_CRCA signals;
3) wait the equivalent of 32 transfer periods (e.g., 200 ns at fast-160);
4) simultaneously negate REQ and P_CRCA signals;
5) wait 32 the equivalent of transfer periods (e.g., 200 ns at fast-160);
6) set precompensation to negotiated state;
7) negate SEL signal;
8) simultaneously assert and negate REQ and P_CRCA signals at the negotiated transfer period 32 times, (e.g., (2 x 6.25) x 32 = 400 ns at fast-160);
9) ~~simultaneously assert and~~ negate REQ and P_CRCA ~~signals~~ for at least the ~~negotiated rate for 64~~ equivalent of 16 transfer ~~periods,~~ periods (e.g., ~~400~~ 100 ns at fast-160); ~~and~~and
10) ~~on detection of the 8th 0000010011111011b pattern on the~~ the ~~(15-0) signals the~~ target shall begin asserting and negating REQ to indicate to the initiator valid data may be sent. The number of REQ ~~transitions~~ assertions shall not exceed the negotiated offset.

The initiator shall begin the section A ~~section~~ of the training pattern ~~on detection of~~ if it detects the ~~assertion of the~~ SEL ~~signal with MSG true~~ and MSG true, and C/D and I/O ~~false~~false on the first assertion of the REQ signal. The initiator shall transmit the following training pattern:

Start of ~~A section~~section A;
~~1) simultaneously assert ACK, P1, and DB(15-0) signals;~~
1) disable precompensation;
2) simultaneously assert ACK, P1, and DB(15-0) signals within the equivalent of 32 transfer periods of the first REQ assertion (e.g., 200 ns at fast-160);
3) wait the equivalent of 32 transfer periods (e.g., 200 ns at fast-160);
4) simultaneously negate ACK, P1, and DB(15-0) signals;
5) wait the equivalent of 32 transfer periods;
6) set precompensation to negotiated state;
7) simultaneously assert and negate ACK, P1, and DB(15-0) signals at the negotiated ~~rate for 128~~ transfer ~~periods~~period 64 times, (e.g., (2 x 6.25) x 64 = 800 ns at fast-160);

Start of ~~B section~~section B;
1) wait the equivalent of 192 transfer periods from the first assertion of ACK in step 2 of section A (e.g., 1200 ns at fast-160);

2) ~~on~~ keep the ~~128th transfer period negate~~ P1, and DB(15-0) signals negated while continuing to assert and negate ACK at the negotiated ~~rate~~ transfer period for the equivalent of 8 transfer periods (e.g., 50 ns at fast-160);
3) ~~negate~~ keep the ACK signal negated for the equivalent of 8 additional transfer periods;
4) simultaneously assert and negate P1 and DB(15-0) signals at twice the negotiated ~~rate for while asserting and negating ACK at the negotiated rate for 48~~ transfer ~~periods~~ period (i.e., simultaneously repeat a 1100b bit pattern 12 times on each signal) while asserting and negating ACK at the negotiated transfer period 24 times (e.g., (2 x 6,25 ns) x 24 = 300 ns at fast-160); and

Start of ~~C section~~section C;

~~5) assert and negate ACK at the negotiated rate for 128 transfer periods while repeating a 0000010011111011b bit pattern eight times on P1 and DB(15-0).~~

6) assert and negate ACK at the negotiated transfer period 64 times and at the same time assert and negate P1 at twice the negotiated transfer period while repeating a 0000010011111011b bit pattern eight times on each of the DB(15-0) signals (e.g., (2 x 6,25 ns) x 64 = 800 ns at fast-160).

At the completion of the training pattern the initiator continues asserting and negating the ACK signal at the negotiated ~~rate~~ transfer period (e.g., 6,25 ns transfer period at fast-160) and the P1 signal at twice the negotiated ~~rate~~ transfer period (e.g., 12 ns transfer period at fast-160). When the initiator is ready to transfer data and ~~there are outstanding REQs~~ the REQ/ACK offset value is not zero it shall reverse the phase of P1 (see 10.8.4.3).

10.8.4.3 P1 data valid/invalid state transitions

10.8.4.3.1 P1 data valid/invalid state transitions overview

The transmitting device shall indicate the start of a data valid ~~data~~ state by reversing the phase of the P1 signal coincident with a REQ or ACK assertion. This is accomplished by withholding the next transition of P1 at the start of the first two transfer periods of valid data. Beginning with the third valid data word, P1 shall be toggled every two transfer periods, coincident with a REQ or ACK assertion. The minimum duration of the data valid state is two transfer periods, and the data valid state shall consist of an even number of transfer periods.

Anytime the sending device pauses the sending of data, it shall reverse the phase of P1 by withholding the next transition of P1 at the start of the first two transfer periods that have invalid data. Beginning with the third transfer period with invalid data, P1 shall be toggled every two transfer periods until valid data is sent. The data invalid state shall have at least one transition of P1 before changing states. The minimum data invalid time is four ~~data~~ transfer periods. This ensures a maximum run length of three cycles for P1. The data invalid state shall last an even number of transfer periods.

From the data invalid state, the sending device may resume sending data by reversing the phase of P1 again.

P1 has the same transmit setup and hold time requirements as data and shall always be detected by the receiving device on the assertion edge of the delayed clocked REQ or ACK signal.

See figure 59 for examples of how the P1 signal is used to determine when the REQ or ACK transition clocks valid data.

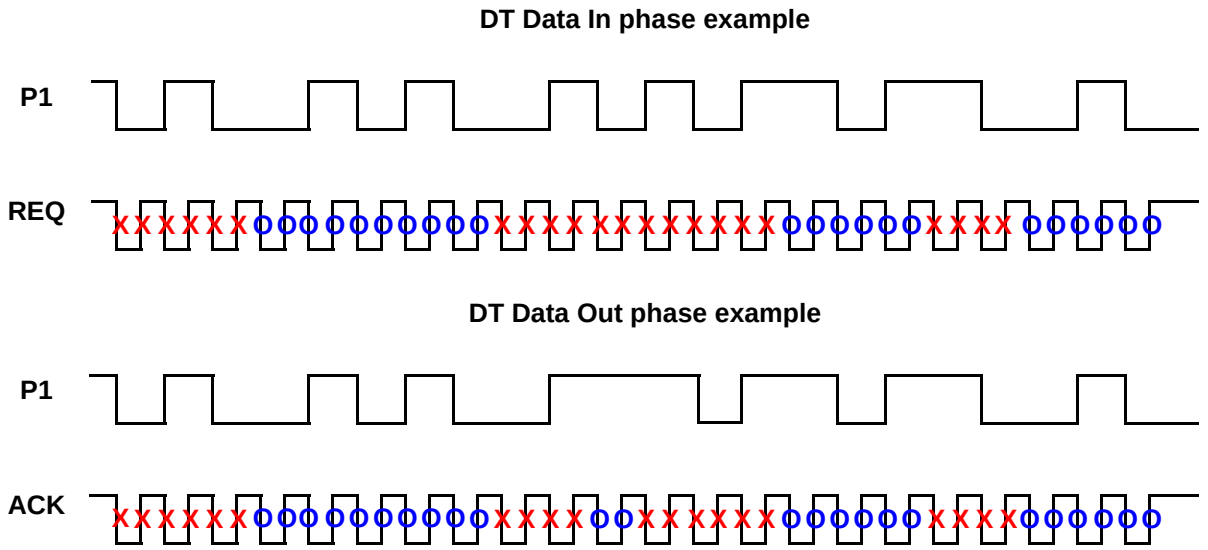


Figure 59 - Usage of P1 to establish data valid and data invalid states

10.8.4.3.2 Starting pacing transfers at end of training pattern

See 10.8.4.2 for the description of starting a data valid state after a training pattern.

10.8.4.3.3 Starting pacing transfers with no training pattern

~~At the start of a DT DATA IN phase without a training pattern (i.e., the SEL signal is not asserted at the start of the DT DATA phase), the target shall begin pacing transfers by:~~

~~1) The target shall being asserting and negating the REQ signal at the negotiated rate 400 ns, or 800 ns if the previous phase was a DT DATA OUT phase, after asserting the MSG and I/O signals and negating the C/D;~~

Before starting the DT DATA IN phase the target shall wait at least two deskew delays after the SEL signal is negated before the first assertion of the REQ signal.

The DT DATA IN phase without training starts on the first assertion of REQ if the SEL is not asserted.

The target shall begin pacing transfers only after meeting all the following:

a) signal restrictions between information transfer phases listed in 10.13;
b) the signal restrictions between a RESELECTION phase and a DT DATA IN phase listed in 10.7.2; or
c) the signal restrictions between a SELECTION phase and a DT DATA OUT phase listed in 10.6.3.

The target shall begin pacing transfers by:

1) simultaneously with the assertion of REQ the target shall begin asserting and negating P1 at twice the negotiated ~~rate~~ transfer period (e.g., 12,5 ns for fast-160);
2) target shall assert and negate P1 ~~for~~ at least ~~16 transfer periods~~ 8 times (e.g., (2 x 6,25 ns) x 8 = 100 ns at fast-160); and

3) the target may establish a data valid state as described in 10.8.4.3.1.

~~At the start of a DT DATA OUT phase without a training pattern (i.e., the SEL signal is not asserted at the start of the DT DATA phase) the following target and initiator actions occur.~~

The DT DATA OUT phase without training starts on the first assertion of REQ if the SEL is not asserted.

The target shall begin pacing transfers only after meeting all the following:

a) signal restrictions between information transfer phases listed in 10.13;
b) the signal restrictions between a RESELECTION phase and a DT DATA IN phase listed in 10.7.2; or
c) the signal restrictions between a SELECTION phase and a DT DATA OUT phase listed in 10.6.3.

The initiator shall begin pacing transfers by:

~~1) The initiator shall being asserting and negating the ACK signal two system deskew delays after detecting the assertion of the MSG signal and negation of the C/D and I/O signals;~~
1) simultaneously with the assertion of ACK the initiator shall begin asserting and negating P1 at twice the negotiated ~~rate~~ transfer period (e.g., 12,5 ns for fast-160);
~~2) initiator shall wait for the assertion of the REQ signal; and~~
3) initiator shall assert and negate P1 at least 8 times (e.g., (2 x 6.25 ns) x 8 = 100 ns at fast-160); and
4) the initiator may establish a data valid state as described in 10.8.4.3.1.

~~The target shall not asset REQ until it has established the data invalid state by detecting valid assertions and negations on the ACK and P1 signals.~~

10.8.4.3.4 Ending pacing transfers

After transmitting the last data word of a DT DATA IN phase the target shall end pacing by waiting for all REQs to be responded to by ACKs then negate the REQ and P1 signals. After the target stops asserting and negating REQ it shall not assert REQ again until ~~a phase change occurs~~the requirements in 10.13 are met.

After transmitting the last data word of a DT DATA OUT phase the initiator shall;

~~a) continue asserting and negating the ACK and P1 signals until it detects a phase change; and~~
~~b) negate the ACK and P1 signals within 200 ns of detecting a phase change;~~
a) continue asserting and negating the ACK and P1 signals until it detects a change to the C/D, I/O, or MSG signals; and
b) negate the ACK and P1 signals within 200 ns of detecting a change to the C/D, I/O, or MSG signals;

When the target changes from a DT DATA OUT phase to any other phase it shall wait at least ~~800 ns~~ a bus settle delay plus a data release delay before asserting REQ and shall ignore any ACK transitions for at least ~~800 ns~~ a bus settle delay plus a data release delay after transitioning ~~any phase~~ the C/D, I/O, or MSG signals.

10.8.4.4 Paced information unit transfer

Editors Note 4 - GOP: This section needs a close looking at. It was moved here because of the additional requirements only being able to transfer data during the data valid state and the requirement the data and REQ/ACK transition at the same time as data. Also added in is the flow control/read steaming controls.

Information units shall be transferred on the DT DATA OUT phase and the DT DATA IN phase, and the information units' embedded iuCRC shall be used to detect information unit data errors.

If the I/O signal is true (transfer to the initiator) and the phase of the P1 signal indicates data is valid, to transfer SPI information units the target:

1) shall drive the DB(15-0) signals to their values simultaneous with the next REQ signal assertion;
2) shall hold the DB(15-0) signals valid for a minimum of one transmit hold time;
3) shall drive the DB(15-0) signals to their values simultaneous with the next REQ signal negation; and
4) shall hold the DB(15-0) signals valid for a minimum of one transmit hold time;

If the I/O signal is true (transfer to the initiator), to receive SPI information units the initiator shall:

1) Read the value on the DB(15-0) signals within one receive hold time of the transition of the REQ signal; and
2) respond with an ACK assertion after each REQ assertion/negation pair.

If the I/O signal is false (transfer to the target) and the phase of the P1 signal indicates data is valid, to transfer SPI information units the initiator:

1) Shall wait until detecting a REQ assertion;
2) shall drive the DB(15-0) signals to their values simultaneous with the next ACK signal assertion;
3) shall hold the DB(15-0) signals valid for at least one transmit hold time;
4) shall drive the DB(15-0) signals to their values simultaneous with the next ACK signal negation;
5) shall hold the DB(15-0) signals valid for at least one transmit hold time;

If the I/O signal is false (transfer to the target), to receive SPI information units the target:

1) Shall read the value of the DB(15-0) signals within one receive hold time of the transition of the ACK;

If flow control is enabled and the current SPI data streaming information unit is the last SPI data stream information unit of the stream then;

1) The target shall assert the P_CRCA signal a minimum of a flow control transmit setup time before the end of the last information unit and shall keep the P_CRCA signal asserted for a flow control transmit hold time;
2) The target shall not assert the P_CRCA signal until a minimum of a flow control hold time after the end of the previous information unit; and
3) The target shall negate the P_CRCA signal a minimum of a flow control transmit setup time before the start of the next information unit.

NOTE 40 - The earlier in a SPI data streaming information unit that the target asserts the P_CRCA signal, the better the initiator may manage data pre-fetch.

As a result of a SPI information unit always being an even number of transfers, the REQ and ACK signals are negated both before and after the transmission of the SPI information unit.

Paced information unit transfers ~~are the same as~~ exception handling conditions are ~~unit transfers and are~~ defined in ~~10.8.3.3.2,~~ 10.8.3.3.3, and 10.8.3.3.4.

10.8.4.5 Deskewing

The deskewing technique used in the receiving SCSI device is vendor specific. Any technique that works with the specified training pattern and complies with the specified receive skew compensation timing requirement is allowed. Deskewing shall only be enabled for paced transfers.

10.8.5 Wide data transfer

Wide data transfers shall be used for DT DATA phases. Wide data transfer may be used in the ST DATA phase only if a non-zero wide data transfer agreement is in effect (see 16.3.16 or 16.3.10). These messages determine the use of wide mode by both SCSI devices and establish a data path width to be used during the ST DATA phase.

A wide data transfer of 16-bits may be established. All SCSI devices shall support 8-bit data transfers.

During 8-bit data transfers, all information shall be transferred in bytes across the DB(7-0) and DB(P_CRCA) signals on the SCSI bus. At the receiving SCSI device the DB(15-8) (if present) and DB(P1) (if present) signals are undefined.

During 16-bit wide data transfers, the first and second information bytes for each DATA phase shall be transferred across the DB(7-0) and DB(15-8) signals, respectively, on the SCSI bus. Subsequent pairs of information bytes are likewise transferred in parallel across the SCSI bus (see table 35).

The IGNORE WIDE RESIDUE message may be used to indicate that the last byte of a data field or the data byte of information unit is undefined.

Table 35 - Wide SCSI byte order

Transfer number	SCSI Bus		Data transfer width
	15.....8	7.....0	
1	N/A	W	8-bit
2	N/A	X	
3	N/A	Y	
4	N/A	Z	
1	X	W	16-bit
2	Z	Y	

Note: When transferring consecutive bytes W, X, Y, and Z across the buses, they are transferred as shown above.
This table does not necessarily represent how these bytes are stored in memory.

If the last information byte transferred does not fall on the DB(15-8) signals for a 16-bit wide transfer then the values of the remaining higher-numbered bits are undefined. However, when using parity protection the DB(P1) signal for this undefined byte shall be valid for whatever data is placed on the bus.

10.9 COMMAND phase

10.9.1 COMMAND phase description

The COMMAND phase allows the target to request command information from the initiator.

The target shall assert the C/D signal and negate the I/O and MSG signals during the REQ/ACK handshake(s) of this phase.

A QAS-capable initiator shall wait a minimum of a QAS non-DATA phase REQ (ACK) period to assert ACK after detecting the assertion of REQ.

A QAS-capable initiator shall assert ACK for a minimum of a QAS non-DATA phase REQ (ACK) period and shall keep the command data valid until the negation of ACK.

10.9.2 COMMAND phase exception condition handling

If the target detects one or more parity error(s) on the command bytes received, it may retry the command by switching to the MESSAGE IN phase and sending a RESTORE POINTERS message. The target shall then switch to the COMMAND phase to receive the original command.

If the target does not retry the COMMAND phase or it exhausts its retry limit it shall return CHECK CONDITION status and set the sense key to ABORTED COMMAND and the additional sense code to SCSI PARITY ERROR.

10.10 DATA phase

10.10.1 DATA phase overview

DATA phase is a term that encompasses both the ST DATA phases and the DT DATA phases. ST DATA phase is a term that encompasses both the ST DATA IN phase and ST DATA OUT phase. DT DATA phase is a term that encompasses both the DT DATA IN phase, and the DT DATA OUT phase.

10.10.2 DT DATA IN phase

The DT DATA IN phase allows the target to request that data be sent to the initiator from the target using DT data transfers.

The target shall assert the I/O and MSG signals and negate the C/D signal during the REQ/ACK handshake(s) of this phase.

10.10.3 DT DATA OUT phase

The DT DATA OUT phase allows the target to request that data be sent from the initiator to the target using DT data transfers.

The target shall assert the MSG signal and negate the C/D and I/O signals during the REQ/ACK handshake(s) of this phase.

10.10.4 ST DATA IN phase

The ST DATA IN phase allows the target to request that data be sent to the initiator from the target using ST data transfers.

The target shall assert the I/O signal and negate the C/D and MSG signals during the REQ/ACK

handshake(s) of this phase.

10.10.5 ST DATA OUT phase

The ST DATA OUT phase allows the target to request that data be sent from the initiator to the target using ST data transfers.

The target shall negate the C/D, I/O, and MSG signals during the REQ/ACK handshake(s) of this phase.

10.11 STATUS phase

10.11.1 STATUS phase description

The STATUS phase allows the target to request that a status byte be sent from the target to the initiator.

The target shall assert the C/D and I/O signals and negate the MSG signal during the REQ/ACK handshake of this phase.

A QAS-capable initiator shall wait a minimum of one QAS non-DATA phase REQ (ACK) period to assert ACK after detecting the assertion of REQ.

A QAS-capable initiator shall assert ACK for a minimum of one QAS non-DATA phase REQ (ACK) period.

10.11.2 STATUS phase exception condition handling

If the initiator detects a parity error on the status byte the initiator shall create an attention condition. When the target switches to a MESSAGE OUT phase the initiator should send an INITIATOR DETECTED ERROR message (see 16.3.5) to the target. This message notifies the target that the status byte was invalid.

10.12 MESSAGE phase

10.12.1 MESSAGE phase overview

The MESSAGE phase is a term that references either a MESSAGE IN, or a MESSAGE OUT phase. Multiple messages may be sent during either phase. The first byte transferred in either of these phases shall be either a single-byte message or the first byte of a multiple-byte message. Multiple-byte messages shall be wholly contained within a single MESSAGE phase.

10.12.2 MESSAGE IN phase

The MESSAGE IN phase allows the target to request that message(s) be sent to the initiator from the target.

The target shall assert the C/D, I/O, and MSG signals during the REQ/ACK handshake(s) of this phase.

A QAS-capable initiator shall wait a minimum of one QAS non-DATA phase REQ (ACK) period to assert ACK after detecting the assertion of REQ.

A QAS-capable initiator shall assert ACK for a minimum of one QAS non-DATA phase REQ (ACK) period.

10.12.3 MESSAGE IN phase exception condition handling

If the initiator detects a parity error on any message byte it receives the initiator shall create an attention condition. When the target switches to a MESSAGE OUT phase the initiator shall send a MESSAGE

PARITY ERROR message (see 16.3.6) to the target. This message notifies the target that the message in byte was invalid.

10.12.4 MESSAGE OUT phase

The MESSAGE OUT phase allows the target to request that message(s) be sent from the initiator to the target. The target invokes this phase in response to the attention condition created by the initiator (see 12.2).

The target shall assert the C/D and MSG signals and negate the I/O signal during the REQ/ACK handshake(s) of this phase. The target shall handshake byte(s) in this phase until the attention condition is cleared, except when rejecting a message.

A QAS-capable initiator shall wait a minimum of one QAS non-DATA phase REQ (ACK) period to assert ACK after detecting the assertion of REQ.

A QAS-capable initiator shall assert ACK for a minimum of one QAS non-DATA phase REQ (ACK) period and shall keep the message data valid until the negation of ACK.

If the target receives all of the message byte(s) successfully (i.e. no parity errors), it shall indicate that it will not retry by changing to any information transfer phase other than the MESSAGE OUT phase and transferring at least one byte. The target may also indicate that it has successfully received the message byte(s) by changing to the BUS FREE phase (e.g. after receiving ABORT TASK SET or TARGET RESET messages).

10.12.5 MESSAGE OUT phase exception condition handling

If the target detects one or more parity error(s) on the message byte(s) received, it may request a retry of the message(s) by asserting the REQ signal after detecting the attention condition has been cleared and prior to changing to any other phase. The initiator, upon detecting this condition, shall resend all of the previous message byte(s) in the same order as previously sent during this phase. When resending more than one message byte, the initiator shall re-establish the attention condition as described in 12.2.

If the target does not retry the MESSAGE OUT phase or it exhausts its retry limit it may;

a) return CHECK CONDITION status and set the sense key to ABORTED COMMAND and the additional sense code to MESSAGE ERROR or;
b) indicate a protocol error by performing an unexpected bus free.

The target may act on messages as received as long as no parity error is detected and may ignore all remaining messages sent under one attention condition after a parity error is detected. When a sequence of messages is resent by an initiator because of a target detected parity error, the target shall not act on any message that it acted on the first time received.

10.13 Signal restrictions between phases

When the SCSI bus is between two information transfer phases, the following restrictions shall apply to the SCSI bus signals:

a) The BSY and ACK signals shall not change.
b) ~~The BSY, SEL, and ACK signals~~ If paced transfers are disabled the SEL signal shall not change.
c) The REQ signal shall not change until it is asserted to qualify the start of a new phase.
d) The C/D, I/O, MSG, DATA BUS, DB(P_CRCA), and DB(P1) signals may change.
e) If paced transfers are enable the SEL signal may change.
f) When switching the DATA BUS or P_CRCA signal direction from out (initiator driving) to in (target driving), the target shall delay driving the DATA BUS, DB(P_CRCA), and/or DB(P1) by at least one

data release delay plus one bus settle delay after asserting the I/O signal and the initiator shall release the DATA BUS, DB(P_CRCA), and/or DB(P1) no later than one data release delay after the transition of the I/O signal to true. When switching the DATA BUS, DB(P_CRCA), and/or DB(P1) direction from in (target driving) to out (initiator driving), the target shall release the DATA BUS, DB(P_CRCA), and/or DB(P1) no later than one system deskew delay after negating the I/O signal. The initiator shall assert the DATA BUS, DB(P_CRCA), and/or DB(P1) no sooner than one system deskew delay after the detection of the negation of the I/O signal.

g) The P_CRCA signal direction may switch direction while the DATA BUS and/or DB(P1) does not (e.g., changing from COMMAND phase to DT DATA OUT phase). When switching the P_CRCA signal direction from out (initiator driving) to in (target driving), the target shall delay driving the P_CRCA by at least one data release delay plus one bus settle delay after negating the C/D signal and the initiator shall release the P_CRCA signal no later than one data release delay after the transition of the C/D signal to false. When switching the P_CRCA signal direction from in (target driving) to out (initiator driving), the target shall release the P_CRCA signal no later than one system deskew delay after asserting the C/D signal. The initiator shall negate the P_CRCA signal no sooner than one system deskew delay after the detection of the assertion of the C/D signal.

h) The ATN and RST signals may change as defined under the descriptions for the attention condition (see 12.2) and hard reset (see 12.3).

11 DATA BUS protection

11.1 DATA BUS protection overview

The DB(P_CRCA) signal and the DB(P1) signal are used to generate parity or control the transfer of pCRC information on the DATA BUS.

11.2 ST DATA BUS protection (parity)

For ARBITRATION phase the DB(P_CRCA) and DB(P1) signals shall not be checked for parity errors.

For SELECTION and RESELECTION phases valid parity is determined by rules in table 36.

.

Table 36 - Parity checking rules for SELECTION and RESELECTION phases

Action	Condition
Check for odd parity on:	**If at least one bit is active on:**
DB(7-0,P_CRCA)	DB(15-0,P_CRCA,P1)
DB(15-8,P1)	DB(15-8,P1)

NOTE 41 - These rules are necessary to permit interoperation of SCSI devices with different DATA BUS widths. For example, if an 8-bit SCSI device selects a 16-bit SCSI device, the 16-bit SCSI device observes invalid parity on the upper 8 bits of the DATA BUS.

For COMMAND, MESSAGE, and STATUS phases the DB(P_CRCA) signal shall indicate odd parity for DB(7-0). The DB(P1) signal shall not be checked.

For ST DATA phases the DB(P_CRCA) signal shall indicate odd parity for DB(7-0). If 8-bit transfers are enabled the DB(P1) signal shall not be checked. If 16-bit data transfers are enabled the DB(P1) signal shall indicate odd parity for DB(15-8). If 16-bit transfers are enabled and the last information byte transferred does not fall on the DB(15-8) signals DB(P1) shall be valid for whatever data is placed on the bus.

Parity protection is not enabled during DT DATA phases.

11.3 DT DATA BUS protection (CRC)

11.3.1 DT DATA BUS protection (CRC) overview

When pCRC protection or iuCRC protection are enabled the error detecting code is a 32-bit (four byte) Cyclic Redundancy Check (CRC), referred to as CRC-32. It is also used by several other device I/O standards. Four CRC bytes are transferred with data to increase the reliability of data transfers

11.3.2 Error detection capabilities

The CRC detects all single bit errors, any two bits in error, or any combination of errors within a single 32-bit range.

11.3.3 Order of bytes in the CRC field

Figure 60 shows how transmitted data is used to calculate the CRC and how the CRC information is then

transmitted.

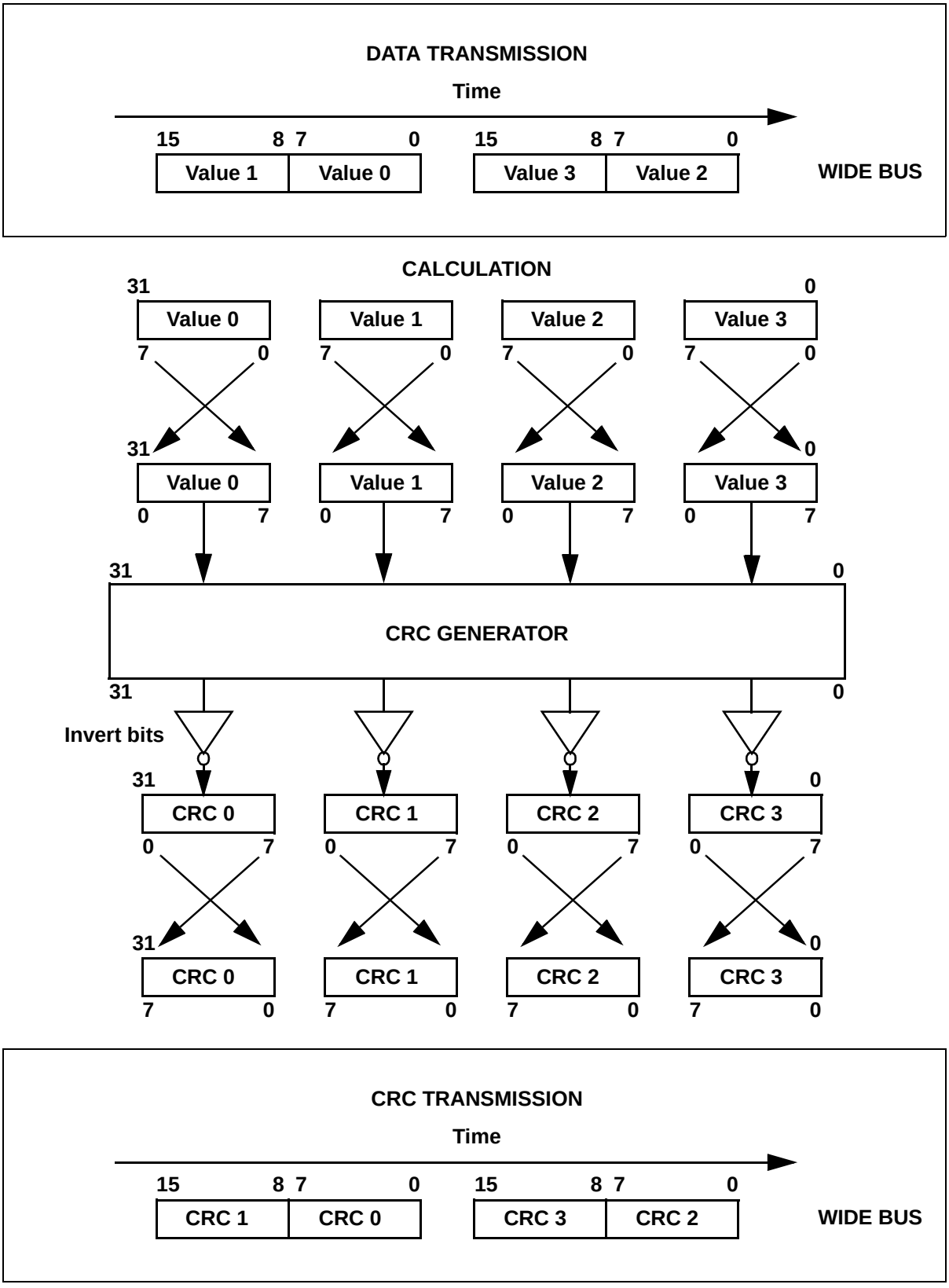


Figure 60 - CRC generation and transmission

11.3.4 CRC generation and checking

The 32-bit generator polynomial used is:

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

This equals 104C11DB7h.

The remainder is generated by dividing the bytes in the data and pad fields by the generator polynomial, modulo two. The remainder is generated 32 bits at a time.

The remainder is initialized to all ones (FFFFFFFFh). This is the seed value. It is reloaded at the beginning of each DT DATA phase and after each CRC is generated/checked.

~~Bytes are bit reversed prior to generating the remainder and the bytes of the remainder are bit reversed prior to forming the CRC field.~~

~~The CRC field is the ones complement of the bit reversed remainder.~~

The data transferred on the asserting edge of a REQ or an ACK is bit-reversed and becomes the most significant 16 bits of the CRC generator's input. The data transferred on the negating edge of a REQ or an ACK is bit-reversed and becomes the least significant 16 bits of the CRC generator's input.

The most significant 16 bits of the CRC generator's output are bit reversed and the one's complement of this result forms the portion of the CRC field that is transferred on the asserting edge of the CRC REQ or ACK. The least significant 16 bits of the CRC generator's output are bit reversed and the one's complement of this result forms the portion of the CRC field that is transferred on the negating edge of the CRC REQ or ACK.

A unique remainder is generated by an error free data group. The unique remainder polynomial of an error free group is:

$$x^{31} + x^{30} + x^{26} + x^{25} + x^{24} + x^{18} + x^{15} + x^{14} + x^{12} + x^{11} + x^{10} + x^8 + x^6 + x^5 + x^4 + x^3 + x + 1$$

This equals C704DD7Bh.

11.3.5 Test cases

For a 32-byte transfer of all 00h, the CRC transferred across the SCSI bus is: 55ADh, 190Ah.

For a 32-byte transfer of all FFh, the CRC transferred across the SCSI bus is: AB0Bh, FF6Ch.

For a 32-byte transfer of an incrementing pattern from 00h to 1Fh, the CRC transferred across the SCSI bus is: 7E8Ah, 9126h.

12 SCSI bus conditions

12.1 SCSI bus conditions overview

The SCSI bus has asynchronous conditions that cause the SCSI device to perform certain actions that may alter the phase sequence.

Furthermore, SCSI devices may not all be powered on at the same time. This standard does not address power sequencing issues. However, each SCSI device, as it is powered on, should perform appropriate internal reset operations and internal test operations. Following a power on to selection time after powering on, SCSI targets should be able to respond with appropriate status and sense data to the TEST UNIT READY, INQUIRY, and REQUEST SENSE commands.

12.2 Attention condition

The attention condition allows an initiator to inform a target that the initiator has a message ready. The target shall honor all valid attention conditions by performing a MESSAGE OUT phase.

The initiator may create an attention condition during the SELECTION phase and during all information transfer phases.

To create an attention condition during the SELECTION phase following normal arbitration, the initiator shall assert the ATN signal at least two system deskew delays before releasing the BSY signal.

To create an attention condition during the SELECTION phase following a QAS, the initiator shall assert the ATN signal at least two system deskew delays before asserting the targets ID on the bus.

To create an attention condition during an information transfer phase, the initiator shall assert the ATN signal at least an ATN transmit setup time before negating the ACK signal. To re-establish an attention condition during a multi-byte MESSAGE OUT retry, the initiator shall assert the ATN signal two system deskew delays before asserting the ACK signal on the first message byte. To clear an attention condition during an information transfer phase, the initiator shall negate the ATN signal at least two system deskew delays before asserting the ACK signal. The initiator shall not negate the ATN signal while the ACK signal is asserted during a MESSAGE OUT phase.

The initiator shall create the attention condition on or before the last information transfer in a bus phase or information unit, for the attention condition to be honored before transition to a new bus phase or information unit. If the initiator does not meet the attention condition setup time, the target may not honor the attention condition until a later bus phase or information unit, which may result in an unexpected action. The initiator shall keep the ATN signal asserted until the target responds to the attention condition.

Once the target has responded to the attention condition by going to MESSAGE OUT phase, the initiator shall keep the attention condition set if more than one message byte is to be transferred. The initiator shall clear the attention condition on the last message byte to be sent. The initiator shall clear the attention condition while transferring the last byte of the messages indicated with a Yes in tables 50, 64, and 69. If the target detects that the initiator failed to meet this requirement, then the target shall go to BUS FREE phase (see 10.2).

A target shall respond to an attention condition with MESSAGE OUT phase as follows:

a) If an attention condition is detected during a COMMAND phase, the target shall enter MESSAGE OUT phase after transferring part or all of the command descriptor block.
b) If an attention condition is detected during a DATA phase, the target shall enter MESSAGE OUT phase at the target's earliest convenience (often, but not necessarily on a logical block boundary). The initiator shall continue REQ/ACK handshakes until it detects a change in the ~~phase change~~C/D, I/O, or MSG signals.

c) If an attention condition is detected during a STATUS phase, the target shall enter MESSAGE OUT phase after the status byte has been acknowledged by the initiator.
d) If an attention condition is detected during a MESSAGE IN phase, the target shall enter MESSAGE OUT phase before it sends another message. This permits a MESSAGE PARITY ERROR message from the initiator to be associated with the appropriate message.
e) If an attention condition is detected during a SELECTION phase the target shall enter MESSAGE OUT phase after that SELECTION phase.
f) If SPI information unit transfers are disabled and an attention condition is detected during a RESELECTION phase, the target shall enter MESSAGE OUT phase after the target has sent its IDENTIFY message for that RESELECTION phase.
g) If the attention condition is detected during an information unit transfer, the target shall enter MESSAGE OUT phase at the completion of the current SPI information unit.
h) If the attention condition is detected between SPI information units the target shall enter MESSAGE OUT phase at the completion of the next SPI information unit.

During a RESELECTION phase the initiator should only create an attention condition to transmit an ABORT TASK SET, ABORT TASK, TARGET RESET, CLEAR TASK SET, DISCONNECT, LOGICAL UNIT RESET, or NO OPERATION message. Other uses may result in ambiguities concerning the nexus.

The initiator shall keep the ATN signal asserted throughout the MESSAGE OUT phase if more than one byte is to be transferred. Unless otherwise specified, the initiator may negate the ATN signal at any time, that does not violate the specified setup and hold times, except it shall not negate the ATN signal while the ACK signal is asserted during a MESSAGE OUT phase. Normally, the initiator negates the ATN signal while the REQ signal is true and the ACK signal is false during the last REQ/ACK handshake of the MESSAGE OUT phase.

12.3 Hard reset

The hard reset is used to clear all SCSI devices from the bus. This condition shall take precedence over all other phases and conditions. Any SCSI device may create the reset condition by asserting the RST signal for a minimum of a reset hold time.

All SCSI devices shall release all SCSI bus signals (except the RST signal) within a bus clear delay of the transition of the RST signal to true. The BUS FREE phase always follows the reset condition.

The effect of the hard reset on tasks that have not completed, SCSI device reservations, and SCSI device operating modes is defined in the SCSI Architecture Model-2 standard.

Environmental conditions (e.g. static discharge) may generate brief glitches on the RST signal. SCSI devices shall not react to glitches on the RST signal that are less than a reset delay. The manner of rejecting glitches is vendor-specific. The bus clear delay following a RST signal transition to true is measured from the original transition of the RST signal, not from the time that the signal has been confirmed. This limits the time to confirm the RST signal to a maximum of a bus clear delay.

12.4 Reset events

12.4.1 Reset events overview

When a SCSI device detects a reset event it shall only initiate an internal hard reset (i.e., the SCSI device shall not assert the RST signal).

12.4.2 Transceiver mode change reset event

When a SCSI device that contains multimode transceivers detects a transceiver mode change from LVD mode to MSE mode it shall cause a reset event. In response to the transceiver mode change reset event,

a target shall create a unit attention condition for all initiators. The sense key shall be set to UNIT ATTENTION, and the additional sense code set to TRANSCEIVER MODE CHANGED TO SE.

When a SCSI device that contains multimode transceivers detects a transceiver mode change from MSE mode to LVD mode it shall cause a reset event. In response to the transceiver mode change reset event, a target shall create a unit attention condition for all initiators. The sense key shall be set to UNIT ATTENTION, and the additional sense code set to TRANSCEIVER MODE CHANGED TO LVD.

Any SCSI device that detects a transceiver mode change shall:

a) set the data transfer width to eight-bit transfer mode
b) set the data transfer mode to asynchronous transfer mode, and
c) set to zero all the PPR protocol options bits (see 16.3.10).

In addition any target that detects a transceiver mode change shall switch to a BUS FREE phase.

13 SCSI bus phase sequences

13.1 SCSI bus phase sequences overview

The order in which phases are used on the SCSI bus follows a prescribed sequence.

During DT DATA phases the target shall not change phases except at data group boundaries or SPI information unit boundaries. If an initiator detects a ~~phase~~ change to the C/D, I/O, or MSG signals within a data group or information unit it shall consider any data transferred for that data group or information unit to have been transferred incorrectly. The initiator shall consider this condition a protocol error and respond accordingly.

A hard reset aborts any phase and is always followed by the BUS FREE phase. Also, any phase may be followed by the BUS FREE phase, but many such instances are exception conditions for initiators (see 10.3).

13.2 Phase sequences for physical reconnection and selection using attention condition with information unit transfers disabled

The allowable sequences for a selection using attention condition and physical reconnection while a transfer agreement is in effect that disables information unit transfers shall be as shown in figure 61.

If a data transfer agreement is in effect that disables information unit transfers (see 16.3.10), the normal progression for selection using attention condition (see 10.6.2) is from the BUS FREE phase to ARBITRATION, from ARBITRATION to SELECTION or RESELECTION, and from SELECTION or RESELECTION to one or more of the information transfer phases (i.e., COMMAND, DATA, STATUS, or MESSAGE). The final information transfer phase is normally the MESSAGE IN phase where a DISCONNECT, or TASK COMPLETE message is transferred, followed by the BUS FREE phase.

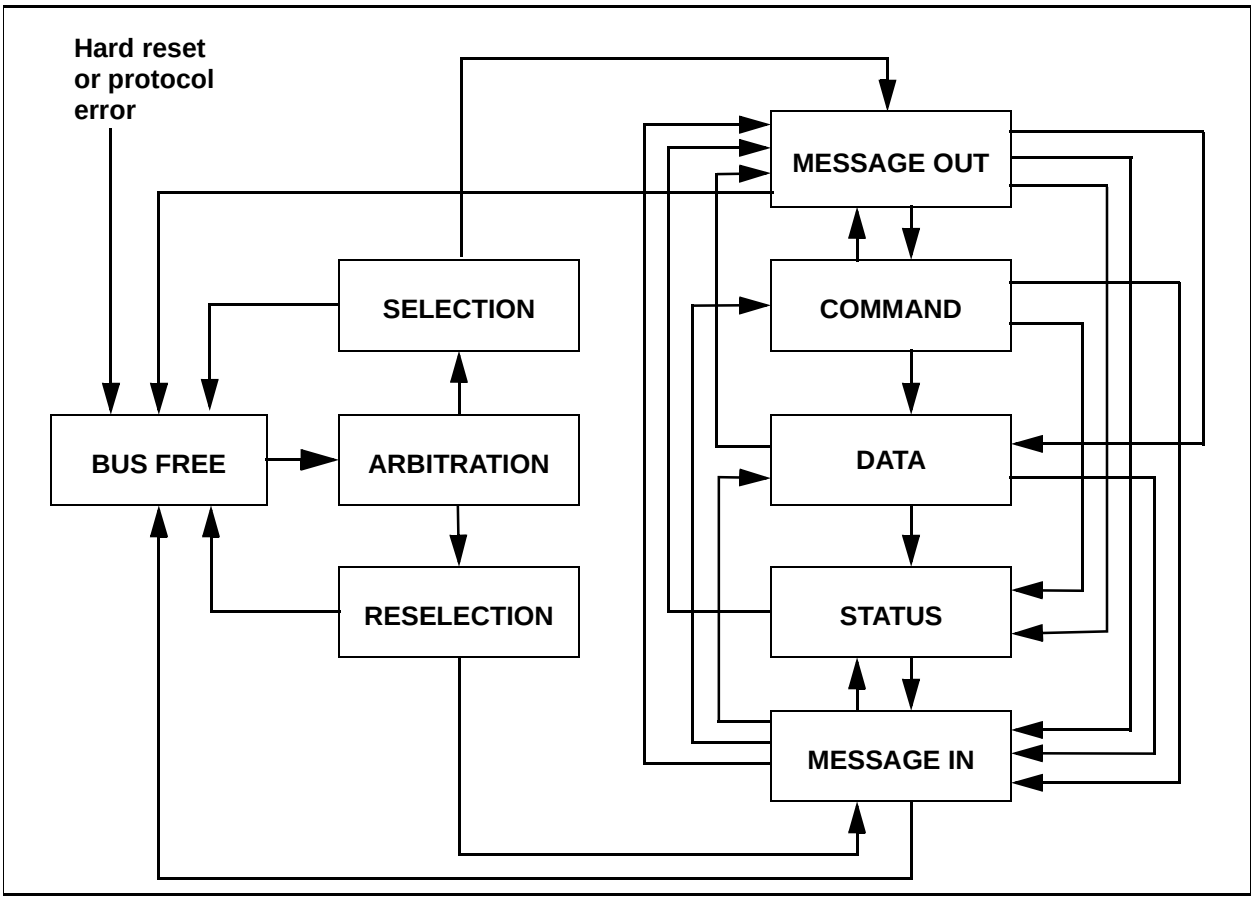


Figure 61 - Phase sequences for selection using attention condition/physical reconnection and information unit transfers disabled

13.3 Phase sequences for selection without using attention condition with information unit transfers disabled

The additional sequences for a selection without using attention condition while a data transfer agreement is in effect that disables information unit transfers shall be as shown in figure 62.

If a data transfer agreement is in effect that disables information unit transfers (see 16.3.10), the normal progression for selection without using attention condition (see 10.6.3) is from the BUS FREE phase to ARBITRATION, from ARBITRATION to SELECTION, from SELECTION to COMMAND phase, from COMMAND phase to DATA phase, from DATA phase to STATUS phase, and from STATUS phase to MESSAGE IN phase where a TASK COMPLETE message is transferred, followed by the BUS FREE phase.

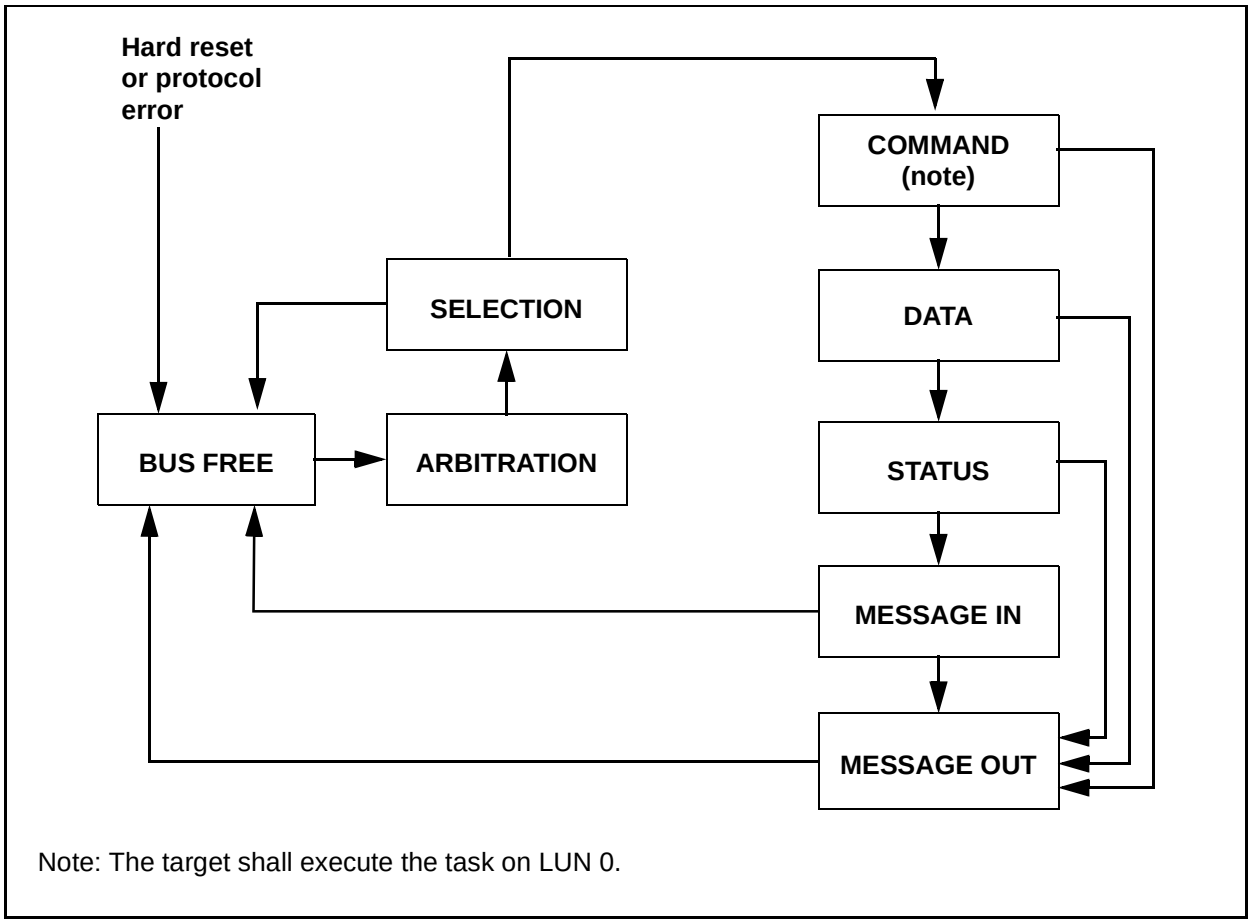
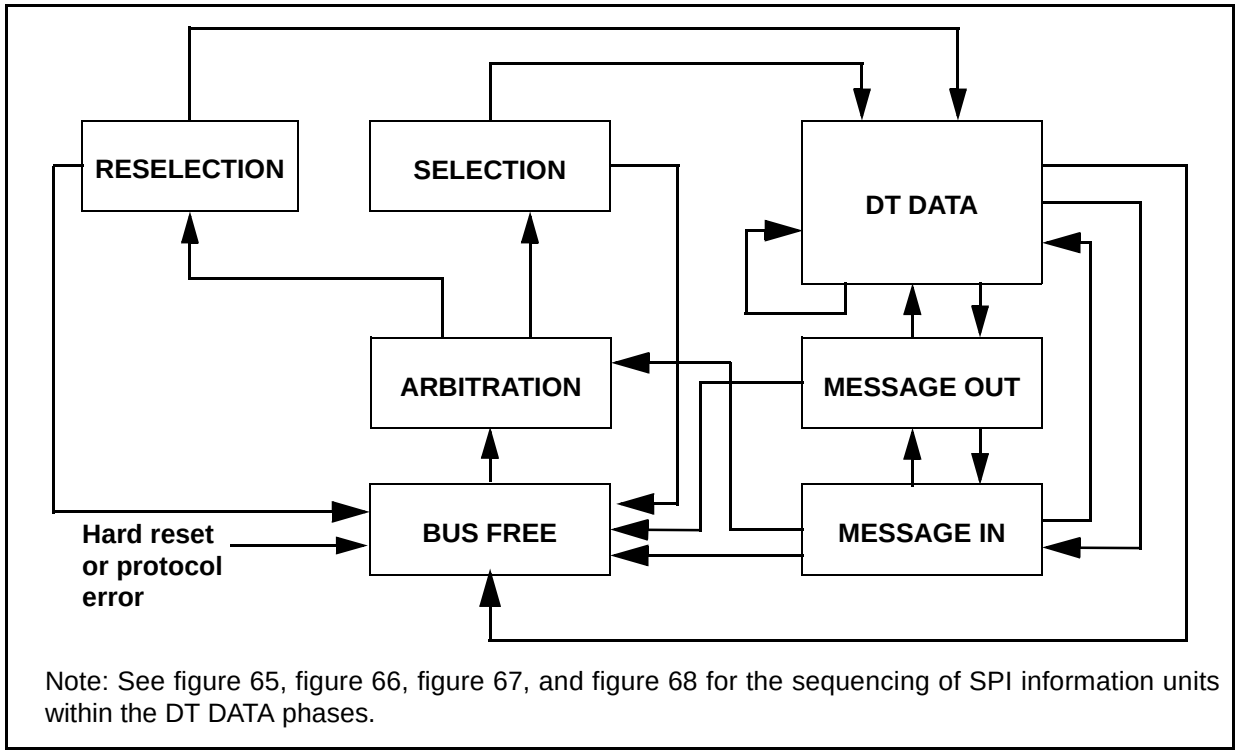


Figure 62 - Phase sequences for selection without using attention condition and information unit transfers disabled

13.4 Phase sequences for selection without using attention condition/physical reconnection with information unit transfers enabled

The sequences for a selection without using attention condition and physical reconnection while a data transfer agreement is in effect that enables information unit transfers shall be as shown in figure 63.

If a data transfer agreement is in effect that enables information unit transfers (see 16.3.10), the normal progression, if QAS is disabled, for selection without using attention condition (see 10.6.3) is from the BUS FREE phase to ARBITRATION, from ARBITRATION to SELECTION or RESELECTION, and from SELECTION or RESELECTION to one or more DT DATA phases. The final DT DATA phase is followed by the BUS FREE phase.



Note: See figure 65, figure 66, figure 67, and figure 68 for the sequencing of SPI information units within the DT DATA phases.

Figure 63 - Phase sequences for selection without using attention condition/physical reconnection and information unit transfers enabled

13.5 Phase sequences for physical selection using attention condition with information unit transfers enabled

The sequences for a selection with attention condition and physical reconnection while a data transfer agreement is in effect that enables information unit transfers shall be as shown in figure 64.

If a data transfer agreement is in effect that enables information unit transfers (see 16.3.10), the normal progression, if QAS is disabled, for selection using attention condition (see 10.6.2.2) is from the BUS FREE phase to ARBITRATION, from ARBITRATION to SELECTION, from SELECTION to MESSAGE OUT, from MESSAGE OUT to MESSAGE IN, and from MESSAGE IN to BUS FREE phase

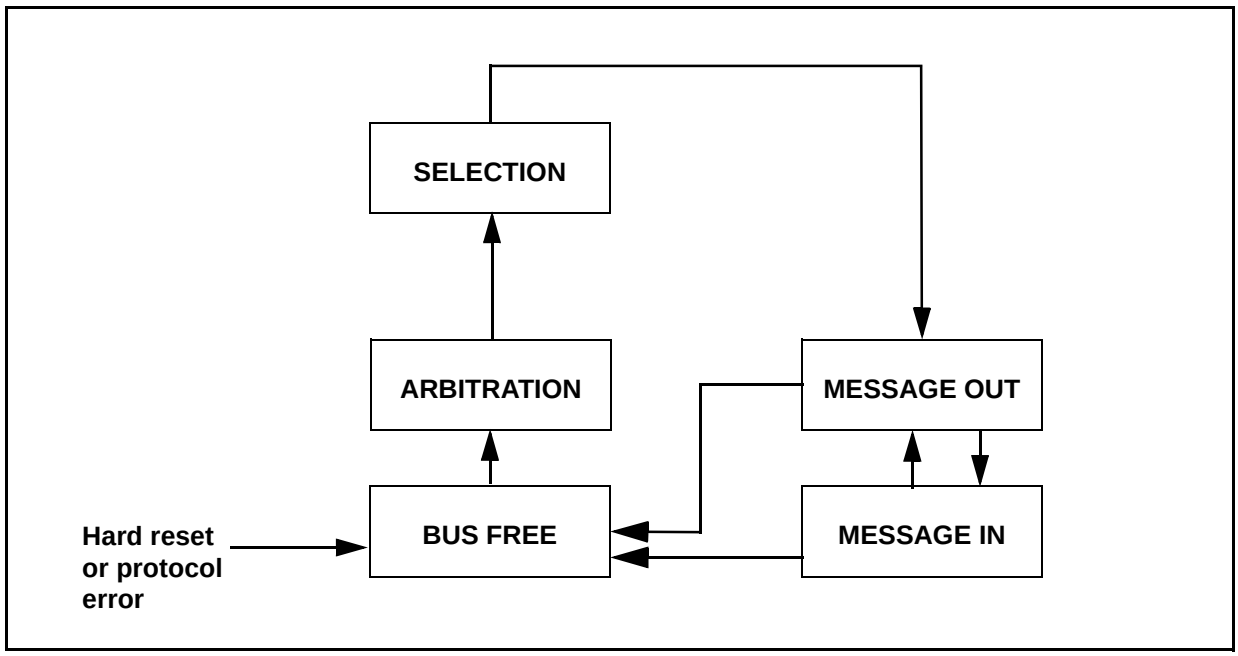


Figure 64 - Phase sequences for selection with attention condition/physical reconnection and information unit transfers enabled

14 SPI information unit sequences

14.1 SPI information unit overview

An information unit transfer transfers data in SPI information units. The order in which SPI information units are transferred within an information unit transfer follows a prescribed sequence. When information unit transfers are enabled only SPI information units shall be transferred within the DT DATA OUT phase and DT DATA IN phase.

The SPI information unit sequences shall be as shown in figure 65, figure 66, figure 67, and figure 68. See figure 63 and figure 62 for the sequencing rules between the DT DATA IN or DT DATA OUT phases and the other phases.

For information on information unit exception handling see 10.8.3.3.3 and 10.8.3.3.4.

The normal progression is from SPI L_Q information unit/SPI command information unit pair(s), to SPI L_Q information unit/SPI data information unit pair(s), to SPI L_Q information unit/SPI status information unit pair(s).

NOTE 42 - An initiator may request a BUS FREE phase by creating an attention condition and sending a DISCONNECT message on the corresponding MESSAGE OUT phase. This allows an initiator to request the target break up a long sequence of SPI L_Q information unit/SPI data information unit pairs into smaller sequences.

When a data transfer agreement is in effect that enables information unit transfers there is no option equivalent to the "physical disconnect without sending a SAVE DATA POINTER message". The initiator shall save the data pointers as soon as the last byte of the last iuCRC for a SPI information unit is transferred. The save shall occur even if the initiator detects an error in the SPI data information unit. ~~If a target retries an operation it shall send a MODIFY DATA POINTERS message then request that the SPI data information unit be transferred again.~~

The target shall not start a new information unit transfer until all previous REQ(s) have been responded to by an equal number of ACK(s) except during a sequence of SPI data stream information units (see 14.3.4).

14.2 Information unit transfer logical operations

SCSI devices using information unit transfers may transfer SPI information units for any number of I/O processes by using logical connects, logical disconnects, and logical reconnects.

If there are no phase changes to a MESSAGE OUT phase or a MESSAGE IN phase then logical disconnects shall only occur at the completion of:

a) each SPI command information unit;
b) each SPI status information unit;
c) each SPI data information unit;
d) any SPI L_Q information unit if the SPI L_Q information unit DATA LENGTH field is zero; and
e) the last SPI data stream information unit.

At completion of those SPI information units the I_T_L_Q nexus becomes an I_T nexus. The I_T nexus remains in place until the target does a physical disconnect or an I_T_L_Q nexus is reestablished by the target transmitting a SPI L_Q information unit.

Logical reconnections occur on the successful target transmission and initiator receipt of a SPI L_Q information unit for an existing I/O process. The logical reconnection reestablishes the I_T_L_Q nexus for that I/O process.

SCSI devices using information unit transfers may receive several commands during an initial connection. This occurs when an initiator uses the multiple command option in the SPI L_Q information unit. For each SPI L_Q received with a multiple command type or a last command type a logical connection occurs and an I_T_L_Q nexus is formed.

If there is a phase change to a MESSAGE OUT phase or a MESSAGE IN phase then there is no logical disconnect and the I_T_L_Q nexus remains in place. If a DT DATA phase follows the message phase then the L_Q portion of the current I_T_L_Q nexus shall be replaced with the L_Q in the next SPI L_Q information unit.

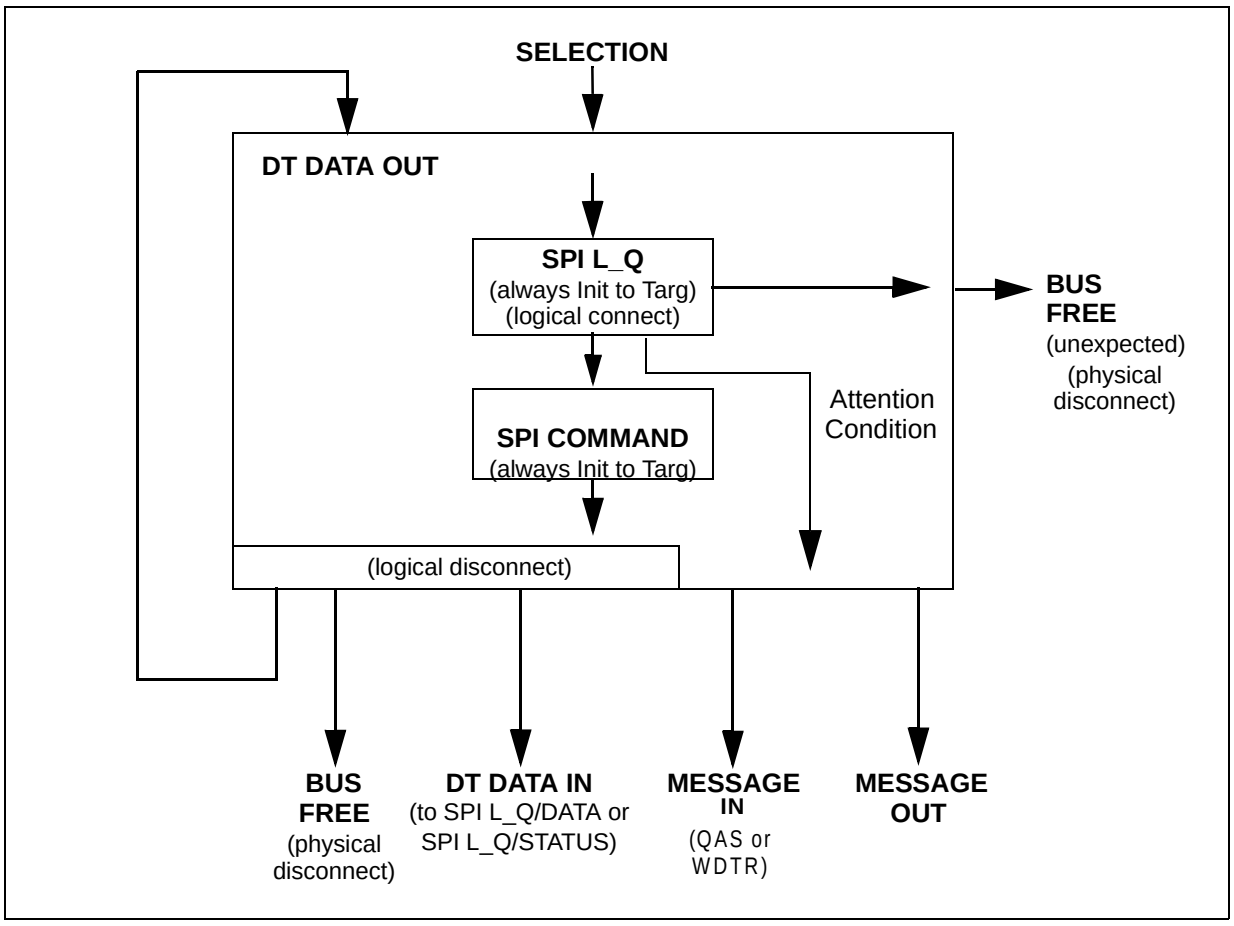


Figure 65 - SPI information unit sequence during initial connection

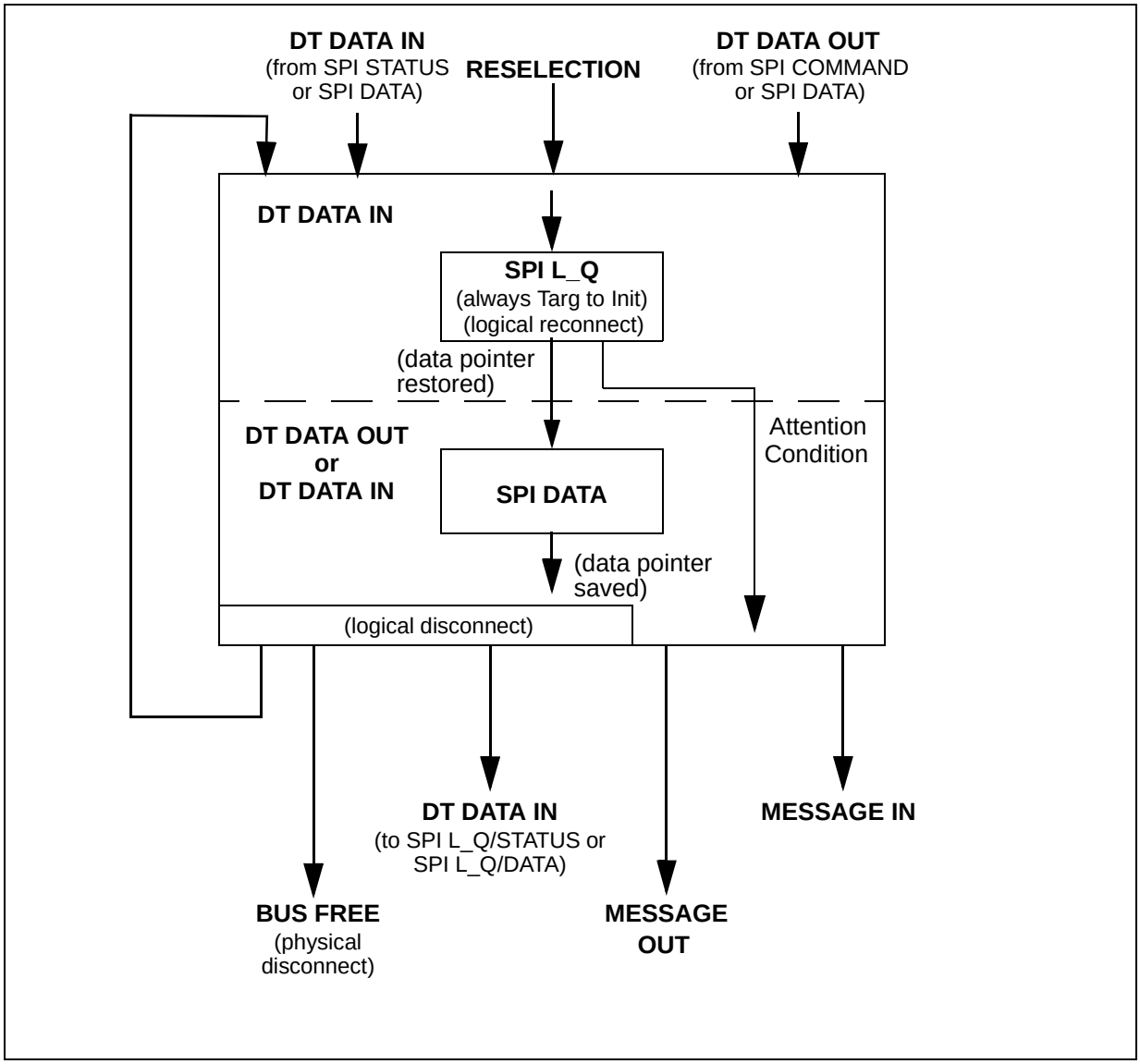


Figure 66 - SPI information unit sequence during data type transfers

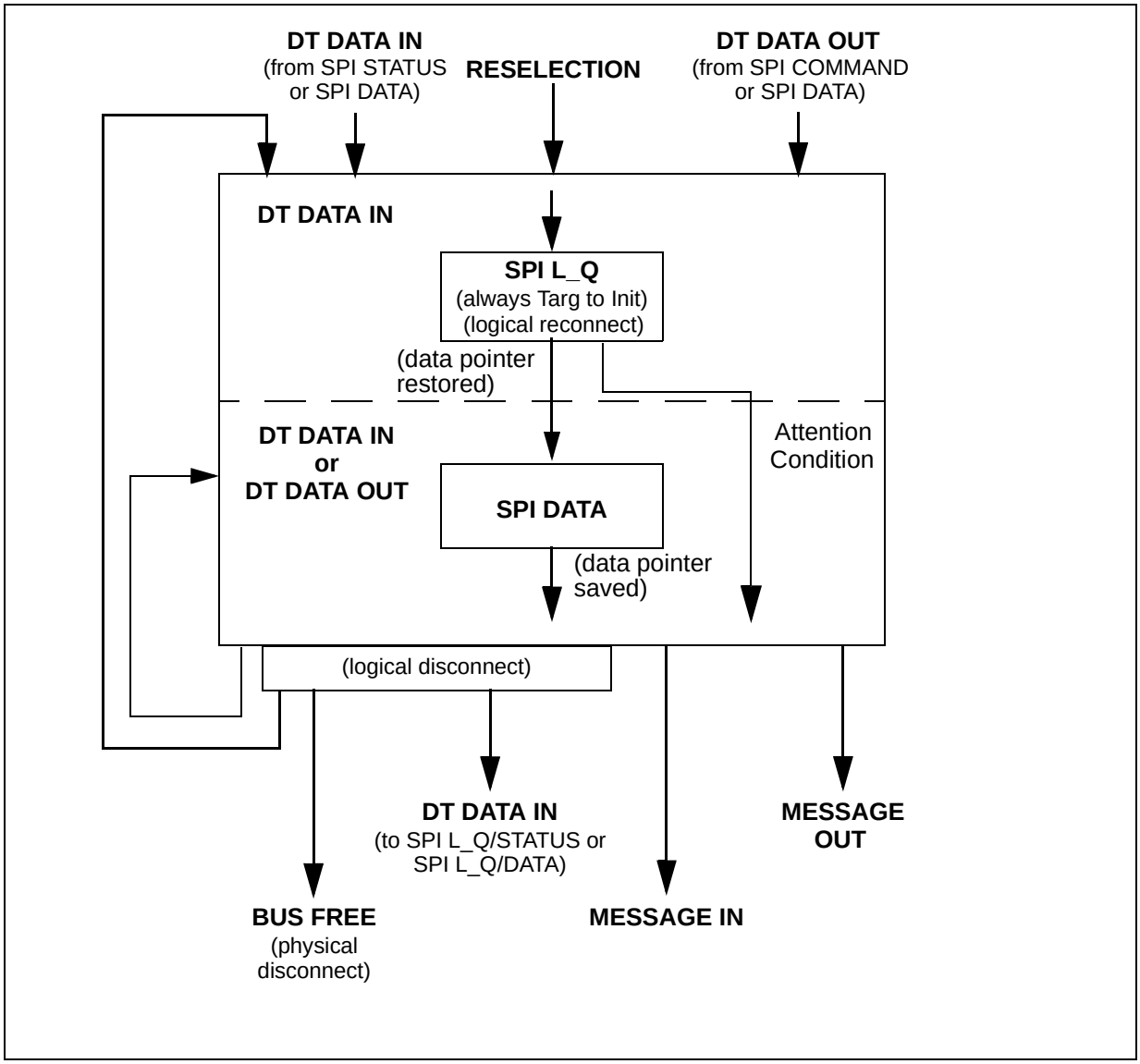


Figure 67 - SPI information unit sequence during data stream type transfers

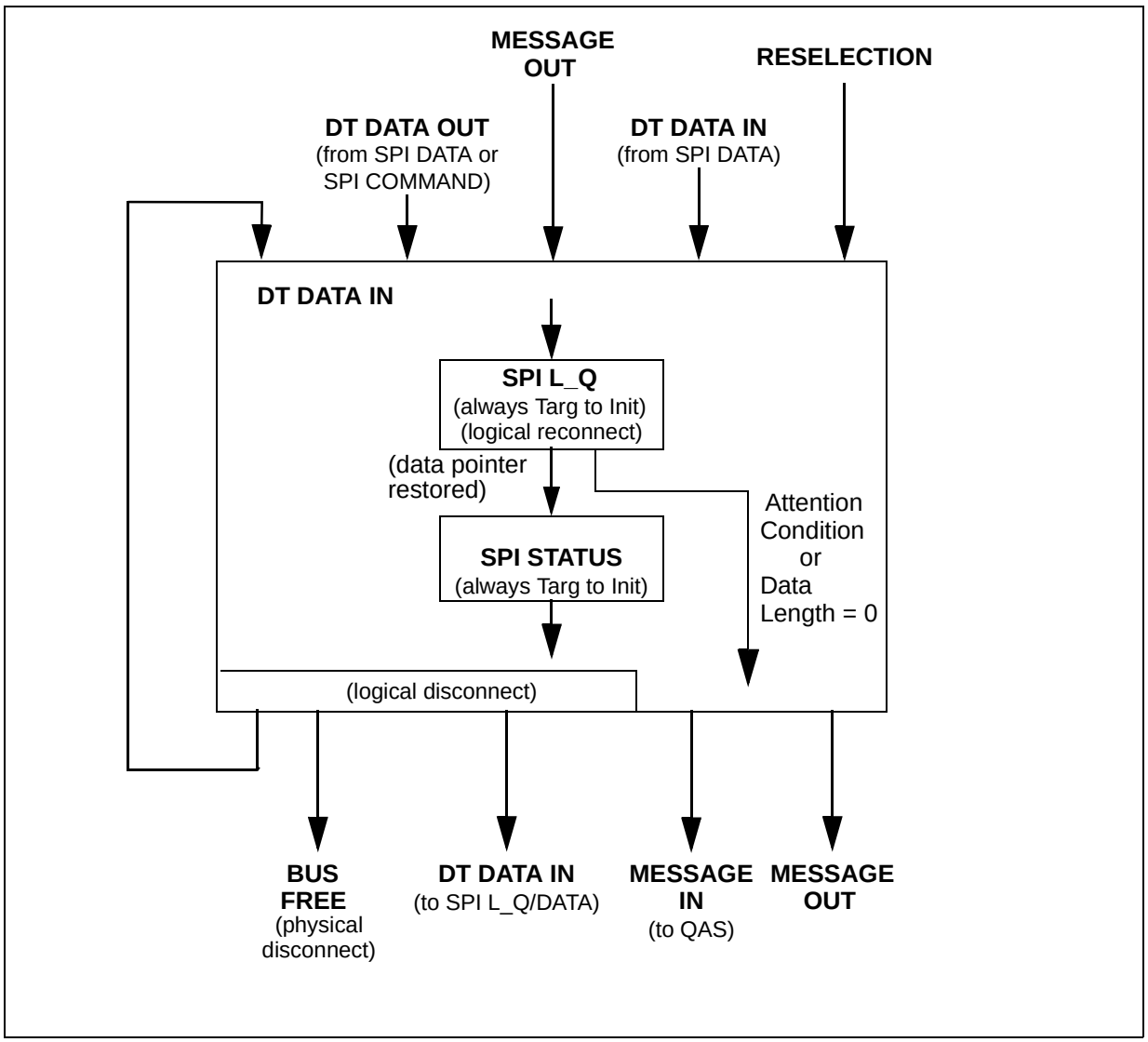


Figure 68 - SPI information unit sequence during status transfers

14.3 SPI information units

14.3.1 SPI command information unit

The SPI command information unit (see table 37) transfers CDBs, task attributes, and task management requests to be performed by a device server.

An initiator shall consider a BUS FREE phase after the transfer of a SPI command information unit to be equivalent to receiving a DISCONNECT message.

If a target does not have the resources to accept a SPI command information unit and the TASK MANAGEMENT FLAGS field equals 00h the target shall transfer all the bytes of the current SPI command information unit and shall discard the transmitted information. After transferring all the SPI command information unit bytes the target shall transmit a SPI status information unit with the status defined in the SCSI Architecture Model-2 standard for this condition. This SPI status information unit may be transferred in the same or a subsequent connection. If the initiator has more commands to send to the target, the initiator shall wait at least until the next selection before those remaining commands may be sent.

If the TASK MANAGEMENT FLAGS field is a supported value not equal 00h the target shall perform the selected task management function before processing any further SPI information units regardless of the command type. On completion of a supported task management function the target shall go to a BUS FREE phase. No SPI status information unit shall be reported for the task management function. If the TASK MANAGEMENT FLAGS field is not a supported value then the task manager shall terminate the task with a GOOD status and the packetized failure code shall be set to TASK MANAGEMENT FUNCTION NOT SUPPORTED. If a task management function fails the task manager shall terminate the task with a GOOD status. The packetized failure code shall be set to TASK MANAGEMENT FUNCTION FAILED.

If the target terminates a SPI L_Q/SPI command information unit pair for one of the following reasons:

a) TASK SET FULL status,
b) BUSY status,
c) CHECK CONDITION due to a SPI command information unit iuCRC error, or
d) a bus free due to a SPI L_Q information unit iuCRC error

it shall have no effect on any other SPI L_Q/SPI command information unit pair beyond those caused by any task management functions contained within the last SPI L_Q/SPI command information unit pair.

Table 37 - SPI command information unit

Bit Byte	7	6	5	4	3	2	1	0
0	RESERVED							
1	RESERVED					TASK ATTRIBUTE		
2	TASK MANAGEMENT FLAGS							
3	ADDITIONAL CDB LENGTH = (n-19)/4						RDDATA	WRDATA
4	CDB							
19								
20	MSB	ADDITIONAL CDB						
n								LSB
n+1	MSB	IUCRC						
n+2								
n+3								
n+4								LSB

The TASK ATTRIBUTE field is defined in table 38.

Table 38 - TASK ATTRIBUTE

Codes	Description
000b	Requests that the task be managed according to the rules for a simple task attribute. (See SAM-2)
001b	Requests that the task be managed according to the rules for a head of queue task attribute. (See SAM-2)
010b	Requests that the task be managed according to the rules for an ordered attribute. (See SAM-2)
011b	Reserved
100b	Requests that the task be managed according to the rules for a automatic contingent allegiance task attribute. (See SAM-2)
101b-111b	Reserved

The TASK MANAGEMENT FLAGS field is defined in table 39. If a task management function fails the task

manager shall terminate the task with a GOOD status. The packetized failure code shall be set to TASK MANAGEMENT FUNCTION FAILED.

Table 39 - TASK MANAGEMENT FLAGS

Codes	Description
00h	Indicates no task management requests for the current task.
01h	The task manager shall abort the task as defined in the ABORT TASK message (see 16.5.2).
02h	The task manager shall abort the task set as defined in the ABORT TASK SET message (see 16.5.3).
04h	The task manager shall clear the task set as defined in the CLEAR TASK SET message (see 16.5.5).
08h	The task manager shall perform a hard reset to the selected logical unit as defined in the LOGICAL UNIT RESET message (see 16.5.6).
20h	The task manager shall perform a hard reset as defined in the TARGET RESET message (see 16.5.7).
40h	The task manager shall perform a clear ACA as defined in the CLEAR ACA message (see 16.5.4).
All other values reserved	The task manager shall terminate the task with a GOOD status. The packetized failure code shall be set to TASK MANAGEMENT FUNCTION NOT SUPPORTED.

The ADDITIONAL CDB LENGTH field contains the length in 4-byte words of the ADDITIONAL CDB field.

The write data (WRDATA) bit and read data (RDDATA) bit are defined in other SCSI protocol standards and shall be ignored by this standard.

The CDB field contains the actual CDB to be interpreted by the addressed logical unit. The CDB field and the task attribute field is not valid and is ignored if the TASK MANAGEMENT FLAGS field is not zero. Any bytes between the end of a 6 byte CDB, 10 byte CDB, or 12 byte CDB and the end of the CDB field shall be reserved.

The ADDITIONAL CDB field contains any CDB bytes beyond those contained within the standard 16 byte CDB field.

The CDB field, ADDITIONAL CDB field, and TASK ATTRIBUTE field are not valid and are ignored if the TASK MANAGEMENT FLAGS field is not zero.

The contents of the CDB and ADDITIONAL CDB fields shall be as defined in the SCSI command standards.

The IUCRC field shall use the algorithm defined in 11.3.

14.3.2 SPI L_Q information unit

The SPI L_Q information unit (see table 40) contains L_Q nexus information for the information unit to that follows, the type of information unit that follows, and the length of information unit that follows. A SPI L_Q information unit shall precede all SPI command information units, SPI multiple command information units, SPI data information units, SPI status information units, and the first of an uninterrupted sequence of SPI data stream information units.

The receipt of an error free (i.e., no iuCRC error) SPI L_Q information unit by an initiator shall cause the initiator to restore the data pointer.

Table 40 - SPI L_Q information unit

Bit Byte	7	6	5	4	3	2	1	0
0	TYPE							
1	RESERVED							
2	MSB			TAG				
3								LSB
4	MSB			LOGICAL UNIT NUMBER				
11								LSB
12	RESERVED							
13	MSB			DATA LENGTH				
14								
15								LSB
16	RESERVED							
17	RESERVED							
18	MSB			IUCRC INTERVAL				
19								LSB
20	MSB			IUCRC				
21								
22								
23								LSB

The TYPE field is defined in table 41. If an initiator receives a type code that is not defined in table 41 that initiator shall follow the procedures defined in 10.8.3.3.3. If a target receives a type code that is not defined in table 41 that target shall follow the procedures defined in 10.8.3.3.4.

Table 41 - TYPE

Codes	Type	Description
01h	Last Command	Sent by an initiator to indicate a SPI command information unit shall follow this SPI L_Q information unit. Indicates the initiator shall not send any more SPI command information units during the current connection. The value of the DATA LENGTH field shall be greater than or equal to 14h and less than or equal to 90h. The IUCRC INTERVAL field shall be set to zero and ignored by the target.
02h	Multiple Command	Sent by an initiator to indicate a SPI command information unit shall follow this SPI L_Q information unit. Indicates the initiator has another SPI L_Q information unit and SPI command information unit during the current connection. The value of the DATA LENGTH field shall be greater than or equal to 14h and less than or equal to 90h. The IUCRC INTERVAL field shall be set to zero and ignored by the target.
04h	Data	Sent by a target to indicate a SPI data information unit shall follow this SPI L_Q information unit. The DATA LENGTH field shall not be set to zero.
05h	Data Stream	Sent by a target to indicate an unspecified number of SPI data stream information unit shall follow this SPI L_Q information unit. The DATA LENGTH field shall not be set to zero.
08h	Status	Sent by a target to indicate a SPI status information unit may follow this SPI L_Q information unit. A length of zero in the DATA LENGTH field shall indicate no SPI status information unit shall follow the SPI L_Q information unit (see 14.3.5). The IUCRC INTERVAL field shall be set to zero and ignored by the target.
F0h - FFh		Vendor specific
All others		Reserved

The TAG field is an 16-bit unsigned integer assigned by the application client and sent to the initiator in the send SCSI command request (see 19.3.2). For more details on tags see 16.4.1.

The LOGICAL UNIT NUMBER field specifies the address of the logical unit of the I_T_L_Q nexus for the current task. The structure of the logical unit number field shall be as defined in the SCSI Architecture Model-2 standard. If the addressed logical unit does not exist, the task manager shall follow the SCSI rules for selection of invalid logical units as defined in the SCSI Primary Commands-2 standard.

The DATA LENGTH field contains the length in bytes of the following information unit(s). For SPI data stream information units the data length field contains the length in bytes of each SPI data stream information unit that follows (i.e., the total number of bytes transferred would equal the data length times the number of SPI data stream information units transferred). The data length shall not include any of the 4 byte iuCRC nor any transmitted pad bytes (e.g., a data length of 509 with a iuCRC interval of zero or greater than 509 would transfer 509 bytes of data plus 3 bytes of pad plus 4 bytes of iuCRC for a total transfer of 516 bytes). The target shall not set the data length to a value that exceeds the maximum burst size as defined in the disconnect-reconnect page (see 18.1.2).

The IUCRC INTERVAL field contains the length in bytes of the data to be sent before a iuCRC is transferred. The iuCRC interval length shall not include the 4 byte iuCRC nor any transmitted pad bytes (e.g., an iuCRC interval length of 510 transfer 510 bytes of data plus 2 bytes of pad plus 4 bytes of iuCRC for a total transfer of 516 bytes). The iuCRC interval shall be a multiple of two (i.e., odd numbers are not allowed). If the iuCRC interval is equal to zero or is greater than or equal to the data length only one iuCRC shall occur at the end of the SPI information unit.

The IUCRC field shall use the algorithm defined in 11.3.

14.3.3 SPI data information unit

The SPI data information unit (see table 42) contains data.

The detection of a BUS FREE phase following a SPI data information unit by an initiator shall be equivalent to the initiator receiving a DISCONNECT message.

The detection of a QAS REQUEST message following a SPI data information unit by an initiator shall be equivalent to the initiator receiving a DISCONNECT message.

Table 42 - SPI data information unit

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)							
				DATA				
n								(LSB)
n+1	MSB							
n+2								
				IUCRC				
n+3								
n+4								LSB

The DATA field may contain any type of information (e.g., parameter lists, mode pages, user data, etc.).

The IUCRC field shall use the algorithm defined in 11.3. If the IUCRC INTERVAL field of the SPI L_Q information unit contains a value greater than zero and less than the data length then there is an IUCRC field at each iuCRC interval in addition to the iuCRC shown in table 42. These additional IUCRC fields are not shown in table 42.

14.3.4 SPI data stream information unit

The SPI data stream information unit (see table 43) contains data.

~~SPI data stream information units shall only be transferred during DT DATA OUT phases.~~

~~All the SPI data stream information units transferred after a SPI L_Q information unit with a type of data stream shall be the size indicated in the DATA LENGTH field of the SPI L_Q information unit. The sequence of SPI data stream information units shall end with a phase change on a SPI data stream information unit boundary.~~

~~If during a sequence of SPI data stream information units an initiator detects a REQ transition after~~

~~transmitting the last iuCRC for a SPI data stream information unit that initiator shall transmit the next SPI data stream information unit.~~

All the SPI data stream information units transferred after a SPI L_Q information unit with a type of data stream shall be the size indicated in the DATA LENGTH field of the SPI L_Q information unit.

If the data transfer size is not a multiple of the data length, the target shall end the stream at a data length boundary and shall send a new SPI L_Q with a smaller data length to finish the data transfer. The new SPI L_Q may or may not be sent during the current physical connection.

During write streaming the sequence of SPI data stream information units shall end with any change to the C/D, I/O, or MSG signals on a SPI data stream information unit boundary. If during write streaming SPI data stream information units an initiator detects a REQ transition after transmitting the last iuCRC for a SPI data stream information unit that initiator shall transmit the next SPI data stream information unit.

During read streaming the sequence of SPI data stream information units shall end when the P_CRCA signal is asserted before the end of the current SPI data stream information unit boundary. If during the last SPI data stream information unit the P_CRCA signal was asserted and initiator detects a REQ transition after receiving the last iuCRC for a SPI data stream information unit that initiator shall receive the next SPI data stream information unit. If during the last SPI data stream information unit the P_CRCA signal was negated and initiator detects a REQ transition after receiving the last iuCRC for a SPI data stream information unit that initiator shall receive a L_Q information unit.

If during a sequence of SPI data stream information units an initiator detects ~~a phase~~ any change to the C/D, I/O, or MSG signals after transmitting or receiving the last iuCRC for a SPI data stream information unit that initiator shall consider the current I/O process to be logically disconnected or in the case of detecting a BUS FREE phase or a MESSAGE IN phase to be physically disconnected.

The detection of a BUS FREE phase following a SPI data stream information unit by an initiator shall be equivalent to the initiator receiving a DISCONNECT message.

The detection of a QAS REQUEST message following a SPI data stream information unit by an initiator shall be equivalent to the initiator receiving a DISCONNECT message.

To end a sequence of SPI data stream information units an initiator may request a disconnect by establishing an attention condition. The initiator shall continue to transfer or receive data, pad bytes (if any), and iuCRC(s) until the target changes to ~~a new~~ the MESSAGE IN phase.

An initiator may end a sequence of SPI data stream information units on the current SPI data stream information unit boundary by establishing an attention condition before the number of words left to transfer or receive in the current SPI data stream information unit is less than the negotiated offset. In the event that the SPI data stream information unit size is smaller than the negotiated offset the target may continue the sequence of SPI data stream information units across two SPI data stream information units but not three.

During a sequence of SPI data stream information units the offset count is not required to go to zero at the boundary of any SPI data stream information unit if the next SPI information unit is a SPI data stream information ~~unit for the same I_T_L_Q nexus~~unit.

Table 43 - SPI data stream information unit

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)			DATA				
n								(LSB)
n+1	MSB							
n+2				IUCRC				
n+3								
n+4								LSB

The DATA field may contain any type of information (e.g., parameter lists, mode pages, user data, etc.).

The IUCRC field shall use the algorithm defined in 11.3. If the IUCRC INTERVAL field of the SPI L_Q information unit contains a value greater than zero and less than the data length then there is an IUCRC field at each iuCRC interval in addition to the iuCRC shown in table 43. These additional IUCRC fields are not shown in table 43.

14.3.5 SPI status information unit

The SPI status information unit (see table 44) contains the completion status of the task indicated by the preceding SPI L_Q information unit. The target shall consider the SPI status information unit transmission to be successful when there is no attention condition on the transfer of the information unit.

If a task completes with a GOOD status, a SNSVALID bit of zero, and a RSPVALID bit of zero then the target shall set the DATA LENGTH field in the SPI L_Q information unit (see 14.3.2) to zero.

Table 44 - SPI status information unit

Bit Byte	7	6	5	4	3	2	1	0
0	RESERVED							
1	RESERVED							
2	RESERVED				RESERVED FOR FCP		SNSVALID	RSPVALID
3	STATUS							
4	(MSB)	SENSE DATA LIST LENGTH (n-m)						
7								(LSB)
8	(MSB)	PACKETIZED FAILURES LIST LENGTH (m-11)						
11								(LSB)
12	(MSB)	PACKETIZED FAILURES						
m								(LSB)
1+m	(MSB)	SENSE DATA						
n								(LSB)
n+1	MSB	IUCRC						
n+2								
n+3								
n+4								LSB

A sense data valid bit (SNSVALID) of zero indicates the sense data list length shall be ignored and no sense data is provided. A SNSVALID bit of one indicates the SENSE DATA LIST LENGTH field specifies the number of bytes in the SENSE DATA field.

If sense data is provided, the sense data valid bit (SNSVALID) shall be set to one and the SENSE DATA LIST LENGTH field shall specify the number of bytes in the SENSE DATA field. The SENSE DATA LIST LENGTH field shall only contain even lengths greater than zero and shall not be set to a value greater than 252.

If no sense data is provided, the sense data valid bit (SNSVALID) shall be set to zero. The initiator shall ignore the SENSE DATA LIST LENGTH field and shall assume a length of zero.

If packetized failure data is provided, the packetized failures valid bit (RSPVALID) shall be set to one and the PACKETIZED FAILURES LIST LENGTH field shall specify the number of bytes in the PACKETIZED FAILURES field. The PACKETIZED FAILURES LIST LENGTH field shall contain a length of 4. Other lengths are reserved for future standardization.

If no packetized failure data is provided, the packetized failures valid bit (RSPVALID) shall be set to zero. The initiator shall ignore the PACKETIZED FAILURES LIST LENGTH field and shall assume a length of zero.

The STATUS field contains the status of a task that completes. See the SCSI Architecture Model-2 standard

for a list of status codes.

The PACKETIZED FAILURES field (see table 45) contains information describing the packetized failures detected during the execution of a task. The PACKETIZED FAILURES field shall contain valid information if the target detects any of the conditions described by the packetized failure code (see table 46).

Table 45 - PACKETIZED FAILURES **field**

Bit Byte	7	6	5	4	3	2	1	0
0	RESERVED							
1	RESERVED							
2	RESERVED							
3	PACKETIZED FAILURE CODE							

The PACKETIZED FAILURE CODE field is defined in table 46.

Table 46 - PACKETIZED FAILURE CODE

Codes	Description
00h	NO FAILURE or TASK MANAGEMENT FUNCTION COMPLETE.
01h	Reserved
02h	SPI COMMAND INFORMATION UNIT FIELDS INVALID
03h	Reserved
04h	TASK MANAGEMENT FUNCTION NOT SUPPORTED
05h	TASK MANAGEMENT FUNCTION FAILED
06h	INVALID TYPE CODE RECEIVED IN SPI L_Q INFORMATION UNIT
07h-FFh	Reserved

The SENSE DATA field contains the information specified by the SCSI Primary Commands-2 standard for presentation by the REQUEST SENSE command. The proper sense data shall be presented when a SCSI status byte of CHECK CONDITION is presented as specified by the SCSI Primary Commands-2 standard.

The IUCRC field shall use the algorithm defined in 11.3.

15 SCSI pointers

The initiator provides for a set of three pointers for each task, called the saved pointers. The set of three pointers consist of one for the command, one for the data, and one for the status. When a send command service is received from an application client, the task's three saved pointers are copied into the initiator 's set of three active pointers. There is only one set of active pointers in each initiator. The active pointers point to the next command, data, or status byte to be transferred between the initiator and the target. The saved and active pointers reside in the initiator.

The saved command pointer always points to the start of the command descriptor block for the task. The saved status pointer always points to the start of the status area for the task. The saved data pointer points to the start of the data area until the target sends a SAVE DATA POINTER message for the task or after the initiator successfully receives or transmits a SPI data information unit.

In response to the SAVE DATA POINTER message or successful receipt or transmission of a SPI data information unit, the initiator stores the value of the current data pointer into the saved data pointer for that task. The target may restore the active pointers to the saved pointer values for the current task by sending a RESTORE POINTERS message to the initiator. The initiator then copies the set of saved pointers into the set of active pointers. Whenever a target does a physical disconnect from the bus, only the set of saved pointers are retained. The set of active pointers is restored from the set of saved pointers upon a physical reconnection of the task or a successful receipt of a SPI L_Q information unit.

Since the data pointer value may be modified by the target before the task ends, it should not be used to test for actual transfer length because the value may no longer be valid.

16 SCSI messages

16.1 SCSI messages overview

SCSI protocol messages allow communication between an initiator and a target for the purpose of link management. The link management messages used for this purpose are defined within this standard and their use is confined to this standard. Other SCSI protocol messages allow communication between the application client and the task manager for the purpose of task management. The task management functions are defined in the SCSI Architecture Model-2 standard. Messages that convey the task management functions are defined by this standard.

16.2 Message protocols and formats

16.2.1 Message protocol rules

One or more messages may be sent during a single MESSAGE phase, but a message shall not be split between multiple MESSAGE phases.

If information unit transfers are disabled, the first message sent by the initiator after a successful SELECTION phase with an attention condition shall be an IDENTIFY, ABORT TASK SET (see 16.5.3), or TARGET RESET message. If a target receives any other message it shall cause an unexpected bus free by generating a BUS FREE phase (see 10.3).

If the first message is an IDENTIFY message, then it may be followed by other messages, such as the first of a pair of SYNCHRONOUS DATA TRANSFER REQUEST messages. With tagged queuing a task attribute shall follow the IDENTIFY message, then more messages may follow. The IDENTIFY message establishes a logical connection between the initiator and the specified logical unit within the target known as an I_T_L nexus.

If information unit transfers are disabled, after the RESELECTION phase, the target's first message shall be IDENTIFY. This allows the I_T_L nexus to be re-established. Only one logical unit shall be identified for any physical connection or physical reconnection; if a target receives a second IDENTIFY message with a different logical unit number during a physical connection or physical reconnection, it shall cause an unexpected bus free by generating a BUS FREE phase (see 10.3).

If information unit transfers are enabled the target enters a DT DATA phase after the RESELECTION phase as described in figure 63.

All initiators shall implement the mandatory messages tabulated in the "Initiator" column of tables 50, 64, and 69. All targets shall implement the mandatory messages tabulated in the "Target" column of tables 50, 64, and 69.

Whenever an I_T_L nexus is established by an initiator that is allowing physical disconnection, the initiator shall ensure that the active pointers are equal to the saved pointers for that particular logical unit. An implied restore pointers operation shall occur as a result of a RESELECTION phase or a successful receipt of a SPI L_Q information unit.

16.2.2 Message formats

One-byte, Two-byte, and Extended message formats are defined. The first byte of the message determines the format as defined in table 47.

Table 47 - Message format

Code	Message format
00h	One-byte message
01h	Extended messages
02h - 0Ah	One-byte messages
0Bh	Obsolete One-byte messages
0Ch - 0Eh	One-byte messages
0Fh - 10h	Reserved One-byte messages
11h - 13h	Obsolete One-byte messages
14h - 15h	Reserved One-byte messages
16h - 17h	One-byte messages
18h - 1Fh	Reserved One-byte messages
20h - 24h	Two-byte messages
25h - 2Fh	Reserved Two-byte messages
30h - 54h	Reserved
55h	One-byte message
56h - 7Fh	Reserved
80h - FFh	One-byte message (IDENTIFY)

16.2.3 One-byte messages

One-byte messages consist of a single byte transferred during a MESSAGE IN phase or a MESSAGE OUT phase. The code of the byte determines the message that is to be performed as defined in tables 50, 64, and 69.

16.2.4 Two-byte messages

Two-byte messages consist of two consecutive bytes transferred during a MESSAGE IN phase or a MESSAGE OUT phase. The code of the first byte determines the message that is to be performed as defined in tables 50, 64, and 69. The second byte is a parameter byte that is used as defined in the message description.

16.2.5 Extended messages

A value of 01h in the first byte of a message indicates the beginning of a multiple-byte extended message. The minimum number of bytes sent for an extended message is three. All of the extended message bytes shall be transferred in consecutive MESSAGE IN phases or consecutive MESSAGE OUT phases. The extended message format is shown in table 48.

Table 48 - Extended message format

Bit / Byte	7	6	5	4	3	2	1	0
0	EXTENDED MESSAGE (01h)							
1	EXTENDED MESSAGE LENGTH (n-1)							
2	EXTENDED MESSAGE CODE (y)							
3-n	EXTENDED MESSAGE ARGUMENTS							

The EXTENDED MESSAGE LENGTH specifies the length in bytes of the EXTENDED MESSAGE CODE plus the extended message arguments to follow. Therefore, the total length of the message is equal to the EXTENDED MESSAGE LENGTH plus two. A value of zero for the EXTENDED MESSAGE LENGTH indicates 256 bytes follow.

The EXTENDED MESSAGE CODEs are listed in table 49.

Table 49 - Extended message codes

Code	Extended message
00h	MODIFY DATA POINTER
01h	SYNCHRONOUS DATA TRANSFER REQUEST
02h	Reserved
03h	WIDE DATA TRANSFER REQUEST
04h	PARALLEL PROTOCOL REQUEST
05h - FFh	Reserved

The EXTENDED MESSAGE ARGUMENTS are specified within the extended message descriptions (see 16.3.8, 16.3.10, 16.3.14, and 16.3.16).

16.3 Link control messages

16.3.1 Link control message codes

Table 50 - Link control message codes

Code	Support		Message Name	Direction		Clear Attention Condition
	Init	Targ				
04h	O	O	DISCONNECT	In	Out	Yes
80h+	M	O	IDENTIFY	In		n/a
80h+	M	M	IDENTIFY		Out	Not required
23h	O	O	IGNORE WIDE RESIDUE	In		n/a
05h	M	M	INITIATOR DETECTED ERROR		Out	Yes
09h	M	M	MESSAGE PARITY ERROR		Out	Yes
07h	M	M	MESSAGE REJECT	In	Out	Yes
01h,05h,00h	O	O	MODIFY DATA POINTER	In		n/a
08h	M	M	NO OPERATION		Out	Yes
01h,06h,04h	M	M	PARALLEL PROTOCOL REQUEST	In	Out	Yes
55h	O	O	QAS REQUEST	In		n/a
03h	O	O	RESTORE POINTERS	In		n/a
02h	O	O	SAVE DATA POINTER	In		n/a
01h,03h,01h	O	O	SYNCHRONOUS DATA TRANSFER REQUEST	In	Out	Yes
00h	M	M	TASK COMPLETE	In		n/a
01h,02h,03h	O	O	WIDE DATA TRANSFER REQUEST	In	Out	Yes

Key: M=Mandatory support, O=Optional support
In=Target to initiator, Out=Initiator to target
Yes=Initiator shall clear the attention condition before last ACK of the MESSAGE OUT phase.
Not required=Initiator may or may not clear the attention condition before last ACK of the MESSAGE OUT phase (see 12.2).
n/a=Not applicable
Init=initiator, Targ=target
80h+=Codes 80h through FFh are used for IDENTIFY messages

16.3.2 DISCONNECT

The DISCONNECT message is sent from a target to inform an initiator that the target plans to do a physical disconnect by releasing the BSY signal, and that a later physical reconnect is going to be required in order to complete the current task. This message shall not cause the initiator to save the data pointer. The target shall consider the message transmission to be successful when there is no attention condition on the DISCONNECT message.

After successfully sending this message the target shall go to the BUS FREE phase by releasing the BSY

signal.

If information unit transfers are disabled any target that breaks data transfers into one or more physical reconnections shall end each successful data transfer (except possibly the last) with a SAVE DATA POINTER - DISCONNECT message sequence.

If information unit transfers are enabled targets shall not transmit a DISCONNECT message.

This message may also be sent from an initiator to a target to instruct the target to do a physical disconnect. If this option is enabled and a DISCONNECT message is received the target shall either:

a) If information unit transfers are disabled switch to MESSAGE IN phase, send the DISCONNECT message to the initiator (possibly preceded by SAVE DATA POINTER message), and then do a physical disconnect by releasing BSY; or
b) if information unit transfers are enabled, regardless of the QAS mode, do a physical disconnect by releasing BSY.

After releasing the BSY signal, the target shall not participate in another ARBITRATION phase for at least a disconnection delay or the time limit specified in the PHYSICAL DISCONNECT TIME LIMIT mode parameter (see 18.1.2) whichever is greater. If this option is disabled or the target is not able to do a physical disconnect at the time when it receives the DISCONNECT message from the initiator, the target shall respond by sending a MESSAGE REJECT message to the initiator.

16.3.3 IDENTIFY

The IDENTIFY message (see table 51) is sent by either the initiator or the target to establish an I_T_L nexus when information unit transfers are disabled.

Table 51 - IDENTIFY message format

Bit / Byte	7	6	5	4	3	2	1	0
0	IDENTIFY	DISCPRIV	LUN					

The IDENTIFY bit shall be set to one to specify that this is an IDENTIFY message.

A disconnect privilege (DISCPRIV) bit of one specifies that the initiator has granted the target the privilege of doing physical disconnects. A DISCPRIV bit of zero specifies that the target shall not do physical disconnects. This bit is not defined and shall be set to zero when an IDENTIFY message is sent by a target.

The target shall generate a BUSY status (see SCSI Architecture Model-2 standard) for a task not granting a physical disconnect privilege (DISCPRIV bit set to zero) in the IDENTIFY message if:

a) there are any pending tasks, and
b) the target determines that a physical reconnection of one or more pending tasks is required before the current task may be completed.

The LUN field specifies a logical unit number.

Only one logical unit number shall be identified per task. The initiator may send one or more IDENTIFY messages during a task. A second IDENTIFY message with a different value in the LUN field shall not be issued before a BUS FREE phase; if a target receives a second IDENTIFY message with a different value

in this field, it shall cause an unexpected bus free (see 10.3) by generating a BUS FREE phase. Thus an initiator may change the DISCPRIV bit, but shall not attempt to switch to another task. (See the DTDC field of the physical disconnect/reconnect mode page in the 18.1.2 for additional controls over physical disconnection.)

An implied RESTORE POINTERS message shall be performed by the initiator following successful identification of the nexus during the MESSAGE IN phase of a physical reconnection or a successful receipt of a SPI L_Q information unit.

Identification is considered successful during an initial connection or an initiator's physical reconnect when the target detects no error during the transfer of the IDENTIFY message and an optional task attribute message in the MESSAGE OUT phase following the SELECTION phase. See 16.4 for the ordering of the IDENTIFY and task attribute messages. See 10.12.5 for handling target detected errors during the MESSAGE OUT phase.

Identification is considered successful during a target's physical reconnect when there is no attention condition on either the IDENTIFY message or the SIMPLE message for an I_T_L_Q nexus in the MESSAGE IN phase following the RESELECTION phase. See the 16.4 for the ordering of the IDENTIFY and task attribute messages. See 12.2, item d), for handling target detected errors during the MESSAGE IN phase.

16.3.4 IGNORE WIDE RESIDUE

The IGNORE WIDE RESIDUE message (see table 52) shall be sent from a target to indicate that the number of valid bytes sent in the last REQ/ACK handshake data of a DATA IN phase is less than the negotiated transfer width. When information unit transfers are disabled the IGNORE WIDE RESIDUE message shall be sent following that DATA IN phase and prior to any other messages.

If the residual byte contains valid data then the IGNORE WIDE RESIDUE message should not be sent.

Table 52 - IGNORE WIDE RESIDUE message format

Bit / Byte	7	6	5	4	3	2	1	0
0	MESSAGE CODE (23h)							
1	NUMBER OF BYTES TO IGNORE (01h)							

The NUMBER OF BYTES TO IGNORE field indicates the number of invalid data bytes transferred. See table 53 for a definition of the IGNORE field codes.

NOTE 43 - More than one IGNORE WIDE RESIDUE message may occur during a task.

Table 53 - IGNORE **field definition**

Codes	Invalid data bits
	16-bit transfers
00h	Reserved
01h	DB(15-8)
02h	Obsolete
03h	Obsolete
04h-FFh	Reserved

16.3.5 INITIATOR DETECTED ERROR

The INITIATOR DETECTED ERROR message is sent from an initiator to inform a target that an error has occurred that does not preclude the target from retrying the task. The source of the error may either be related to previous activities on the SCSI bus or may be internal to the initiator and unrelated to any previous SCSI bus activity. Although the integrity of the currently active pointers in not assured, a RESTORE POINTERS message or a physical disconnect followed by a reconnect shall cause the pointers to be restored to their defined prior state.

16.3.6 MESSAGE PARITY ERROR

The MESSAGE PARITY ERROR message is sent from the initiator to the target to indicate that it received a message byte with a parity error (see 10.12.5).

In order to indicate its intentions of sending this message, the initiator shall create an attention condition on the message byte that has the parity error. This provides an interlock so that the target is able to determine which message byte has the parity error. If the target receives this message under any other circumstance, it shall signal a catastrophic error condition by going to a BUS FREE phase without any further information transfer attempt (see 10.2).

If the target attempts a retry after receiving the MESSAGE PARITY ERROR message the target shall return to the MESSAGE IN phase before switching to some other phase, the target shall resend the entire message that had the parity error.

16.3.7 MESSAGE REJECT

The MESSAGE REJECT message is sent from either the initiator or target to indicate that the last message or message byte it received was inappropriate or has not been implemented.

In order to indicate its intentions of sending this message, the initiator shall create an attention condition on the message byte that is to be rejected. If the target receives this message under any other circumstance, it shall reject this message.

When a target sends this message, it shall change to MESSAGE IN phase and send this message prior to requesting additional message bytes from the initiator. This provides an interlock so that the initiator is able to determine which message byte is rejected.

After a target sends a MESSAGE REJECT message and if the attention condition is still set, then it shall return to the MESSAGE OUT phase. The subsequent MESSAGE OUT phase shall begin with the first byte of a message.

16.3.8 MODIFY DATA POINTER

The MODIFY DATA POINTER message (see table 54) is sent from the target to the initiator and requests that the signed ARGUMENT be added (two's complement) to the value of the current data pointer. The Enable Modify Data Pointer (EMDP) bit in the disconnect-reconnect mode page (see 18.1.2) indicates whether or not the target is permitted to issue the MODIFY DATA POINTER message.

It is recommended that the target not attempt to move the data pointer outside the range addressed by the command. Initiators may or may not place further restrictions on the acceptable values. Should the target send an ARGUMENT value that is not supported by the initiator, the initiator may reject the value by responding with the MESSAGE REJECT message. In this case, the data pointer is not changed from its value prior to the rejected MODIFY DATA POINTER message.

Table 54 - MODIFY DATA POINTER message format

Bit / Byte	7	6	5	4	3	2	1	0
0	EXTENDED MESSAGE (01h)							
1	EXTENDED MESSAGE LENGTH (05h)							
2	MODIFY DATA POINTER (00h)							
3	(MSB)							
4				ARGUMENT				
5								
6								(LSB)

16.3.9 NO OPERATION

The NO OPERATION message is sent from an initiator in response to a target 's request for a message when the initiator does not currently have any other valid message to send.

For example, if the target does not respond to the attention condition until a later phase and at that time the original message is no longer valid the initiator may send the NO OPERATION message when the target switches to a MESSAGE OUT phase.

16.3.10 PARALLEL PROTOCOL REQUEST

16.3.10.1 PARALLEL PROTOCOL REQUEST message description

PARALLEL PROTOCOL REQUEST messages (see table 55) are used to negotiate a synchronous transfer agreement, a wide data transfer agreement, and set the protocol options between two SCSI devices. Negotiations using this message shall only be initiated by initiators.

Table 55 - PARALLEL PROTOCOL message format

Bit / Byte	7	6	5	4	3	2	1	0
0	EXTENDED MESSAGE (01h)							
1	EXTENDED MESSAGE LENGTH (06h)							
2	PARALLEL PROTOCOL REQUEST (04h)							
3	TRANSFER PERIOD FACTOR							
4	RESERVED							
5	REQ/ACK OFFSET							
6	TRANSFER WIDTH EXPONENT (m)							
7	PCOMP_EN	RTI	RD_STRM	WR_FLOW	RESERVED	QAS_REQ	DT_REQ	IU_REQ

The PERIOD FACTOR field is defined in table 56.

Table 56 - TRANSFER PERIOD FACTOR field

Code	Description
00h-07h	Reserved (note 1)
08h	Transfer period equals 6,25 ns (note 2). This code is only valid if the PROTOCOL OPTIONS field has a value selected that supports ~~paced~~ DT data transfers.
09h	Transfer period equals 12,5 ns (note 3). This code is only valid if the PROTOCOL OPTIONS field has a value selected that supports DT data transfers.
0Ah	Transfer period equals 25 ns (note 4)
0Bh	Transfer period equals 30,3 ns (note 4)
0Ch	Transfer period equals 50 ns (note 5)
0Dh-18h	Transfer period equals the period factor x 4 (note 5)
19h-31h	Transfer period equals the period factor x 4 (note 6)
32h-FFh	Transfer period equals the period factor x 4 (note 7)

note:
1 - Faster timings may be allowed by future SCSI parallel interface standards.
2 - Fast-160 data is latched every 6,25 ns.
3 - Fast-80 data is latched every 12,5 ns.
4 - Fast-40 data is latched every 25 ns or 30,3 ns.
5 - Fast-20 data is latched using a transfer period of less than or equal 96 ns and greater than or equal to 50 ns.
6 - Fast-10 data is latched using a transfer period of less than or equal 196 ns and greater than or equal 100 ns.
7 - Fast-5 data is latched using a transfer period of less than or equal 1 020 ns and greater than or equal to 200 ns

For ST synchronous transfers the REQ/ACK OFFSET is the maximum number of REQ assertions allowed to be outstanding before a corresponding ACK assertion is received at the target. The width of a data transfer may be 1 or 2 bytes depending on the values in the transfer width exponent field.

For DT synchronous transfers the REQ/ACK OFFSET is the maximum number of REQ transitions allowed to be outstanding before a corresponding ACK transition is received at the target. The width of a data transfer shall be 2 bytes.

For ~~ST synchronous~~ paced DT DATA IN transfers the REQ/ACK OFFSET is the maximum number of data valid state REQ assertions (see 10.8.4.3) allowed to be outstanding before a corresponding ACK assertion is received at the target. The width of a data transfer ~~may~~ shall be ~~1 or~~ 2 ~~bytes depending on the values in the transfer width exponent field~~bytes. Each assertion indicates a 32-bit data transfer.

For paced DT ~~synchronous transfer~~ DATA OUT transfers the REQ/ACK OFFSET is the maximum number of REQ ~~transitions~~ assertions allowed to be outstanding before a corresponding data valid state ACK ~~transition~~ assertion is received at the target. The width of a data transfer shall be 2 bytes. Each assertion indicates a 32-bit data transfer.

See 4.8 for an explanation of the differences between DT and ST data transfers.

The REQ/ACK OFFSET value is chosen to prevent overflow conditions in the SCSI device's reception buffer and offset counter. A REQ/ACK OFFSET value of zero shall indicate asynchronous transfer mode and that the PERIOD FACTOR field and the PROTOCOL OPTIONS field shall be ignored; a value of FFh shall indicate unlimited REQ/ACK offset.

The TRANSFER WIDTH EXPONENT field defines the transfer width to be used during DATA IN phases, and DATA OUT phases. The transfer width that is established applies to both SCSI devices. Valid transfer widths are 8 bits (m = 00h) and 16 bits (m = 01h) if all the protocol options bits are zero. The only valid transfer width is 16 bits (m = 01h) if any of the protocol options bits are one. TRANSFER WIDTH EXPONENT field values greater than 01h are reserved.

The protocol options bits (RD_STRM, WR_FLOW, RTI, IU_REQ, DT_REQ, and QAS_REQ~~, and PACE_ON~~) are used by the initiator to indicate the protocol options to be enabled. The target uses the protocol options bits to indicate to the initiator if the requested protocol options are enabled. The target shall not enable any options that were not enabled in the PPR message received from the initiator.

An information units enable request bit (IU_REQ) of zero indicates that information unit transfers shall not be used (i.e., data group transfers shall be enabled) when received from the initiator and that information unit transfers are disabled when received from the target. An IU_REQ bit of one indicates that information unit transfers shall be used when received from the initiator and that information unit transfers are enabled when received from the target. ~~If~~ Each time the IU_REQ bit is changed from the previous agreement (i.e., zero to one or one to zero) as a result of a negotiation the target shall go to a BUS FREE phase on completion of the negotiation.

A DT enable request bit (DT_REQ) of zero indicates that DT DATA phases are to be disabled when received from the initiator and that DT DATA phases are disabled when received from the target. An DT_REQ bit of one indicates that DT DATA phases are to be enabled when received from the initiator and that DT DATA phases are enabled when received from the target.

A QAS enable request bit (QAS_REQ) of zero indicates that QAS is to be disabled when received from the initiator and that QAS is disabled when received from the target. A QAS_REQ bit of one indicates that QAS is to be enabled when received from the initiator and that QAS is enabled when received from the target.

A retain training information bit (RTI) of zero received by a target indicates the initiator does not support saving training information and the target shall respond with the RTI bit set to zero. For negotiated transfer periods greater than 6.25 ns the RTI bit shall be set to zero. A RTI bit of one received by a target indicate the initiator does support saving training information and the target may respond with the RTI bit set one if it supports saving training information.

A write flow control bit (WR_FLOW) of zero indicates that flow control during write streaming shall be disabled. A WR_FLOW bit of one indicates flow control during write streaming shall be enabled.

A ~~pacing transfers enabled~~ read streaming bit (~~PACE_ON~~RD_STRM) of zero indicates ~~that paced transfers~~ read streaming shall be disabled. A ~~PACE_ON~~ RD_STRM bit of one indicates ~~that paced transfers~~ read streaming shall be ~~used on all DT DATA phases. For negotiated transfer periods greater~~ enabled. ~~6.25 ns the PACE_ON bit shall be set to zero. For a negotiated transfer period of 6.25 ns the PACE_ON bit shall be set to one.~~

Not all combinations of the protocol options bits are valid. Only the bit combinations defined in table 57 shall be allowed. All other combinations are reserved.

Table 57 - Valid protocol options bit combinations

RTI	RD_STRM	WR_FLOW	QAS_REQ	DT_REQ	IU_REQ	**Description**
0	0	0	0	0	0	Use ST DATA IN and ST DATA OUT phases to transfer data
0	0	0	0	1	0	Use DT DATA IN and DT DATA OUT phases with data group transfers
0 or if f-160 1	1 or 0	1 or 0	0	1	1	Use DT DATA IN and DT DATA OUT phases with information unit transfers
0 or if f-160 1	1 or 0	1 or 0	1	1	1	Use DT DATA IN and DT DATA OUT phases with information unit transfers and use QAS for arbitration

~~A~~ For negotiated transfer periods equal to 6,25 ns a precompensation enabled bit (~~P_EN~~PCOMP_EN) of zero indicates to the SCSI device that receives the PPR message that it shall disable precompensation on all signals transmitted during DT DATA phases. For negotiated transfer periods greater than 6,25 ns the ~~P_EN~~ PCOMP_EN bit shall be set to zero. A ~~P_EN~~ PCOMP_EN bit of one indicates to the SCSI device that receives the PPR message that it shall enable precompensation on all signals transmitted during DT DATA phases. SCSI devices that support fast-160 shall support a the receipt of a ~~P_EN~~ PCOMP_EN bit set to zero or one.

Unlike other fields and bits in the PPR message the ~~P_EN~~ PCOMP_EN bit is not a negotiated value, rather it instructs the receiving SCSI device as to whether or not precompensation is to be disabled or enabled. Because of this precompensation may be enabled on one of the SCSI devices and disabled on the other SCSI device at the completion a successful PPR negotiation.

A PARALLEL PROTOCOL REQUEST agreement applies to all logical units of the two SCSI devices that negotiated agreement. That is, if SCSI device A, acting as an initiator negotiates a data transfer agreement with SCSI device B (a target), then the same data transfer agreement applies to SCSI devices A and B even if SCSI device B changes to an initiator.

A data transfer agreement only applies to the two SCSI devices that negotiate the agreement. Separate data transfer agreements are negotiated for each pair of SCSI devices. The data transfer agreement only applies to DATA phases and information unit transfers.

A PARALLEL PROTOCOL REQUEST message exchange shall be initiated by an initiator whenever a previously arranged parallel protocol agreement may have become invalid. The agreement becomes invalid after any condition that may leave the parallel protocol agreement in an indeterminate state such as:

a) after a hard reset;
b) after a TARGET RESET message;
c) after a power cycle;
d) after a change in the transceiver mode (e.g., LVD mode to SE mode).

If a target determines that the agreement is in an indeterminate state it shall initiate a negotiation to enter an asynchronous, eight-bit wide data transfer mode with all the protocol options bits set to set to zero. using a WDTR message with the TRANSFER WIDTH EXPONENT set to 00h.

Any condition that leaves the data transfer agreement in an indeterminate state shall cause the SCSI device to enter an asynchronous, eight-bit wide data transfer mode with all the protocol options bits set to

set to zero.

An initiator may initiate a PARALLEL PROTOCOL REQUEST message exchange whenever it is appropriate to negotiate a data transfer agreement. SCSI devices that are currently capable of supporting any of the PARALLEL PROTOCOL REQUEST options shall not respond to a PARALLEL PROTOCOL REQUEST message with a MESSAGE REJECT message.

Renegotiation after every selection is not recommended, since a significant performance impact is likely.

The PARALLEL PROTOCOL REQUEST message exchange establishes an agreement between the two SCSI devices;

a) on the permissible periods and the REQ/ACK offsets for all logical units on the two SCSI devices. This agreement only applies to ST DATA IN phases, ST DATA OUT phases, DT DATA IN phases, and DT DATA OUT phases. All other phases shall use asynchronous transfers;
b) on the width of the data path to be used for DATA phase transfers between two SCSI devices. This agreement only applies to ST DATA IN phases, ST DATA OUT phases, DT DATA IN phases, and DT DATA OUT phases. All other information transfer phases shall use an eight-bit data path; and
c) on the protocol option is to be used.

The initiator sets its values according to the rules above to permit it to receive data successfully. If the target is able to receive data successfully with these values (or smaller periods or larger REQ/ACK offsets or both), it returns the same values in its PARALLEL PROTOCOL REQUEST message, except for the ~~P_EN~~ PCOMP_EN value. If it requires a larger period, a smaller REQ/ACK offset, or a smaller transfer width in order to receive data successfully, it substitutes values in its PARALLEL PROTOCOL REQUEST message as required, returning unchanged any value not required to be changed. If the ~~PACE_ON bit is set to zero~~TRANSFER PERIOD FACTOR contains a value greater than 08h, each SCSI device when transmitting data shall respect the negotiated limits set by the other's PARALLEL PROTOCOL REQUEST message, but it is permitted to transfer data with larger periods, smaller synchronous REQ/ACK offsets, or both. If the ~~PACE_ON bit is set~~ TRANSFER PERIOD FACTOR contains a value equal to ~~one~~08h, each SCSI device when transmitting data shall transmit data at the negotiated transfer period but smaller synchronous REQ/ACK offsets are allowed. The completion of an exchange of PARALLEL PROTOCOL REQUEST messages implies an agreement as shown in table 58.

If the target does not support the selected protocol option it shall clear as many bits as required to set the protocol option field to a legal value that it does support.

Table 58 - PARALLEL PROTOCOL REQUEST messages implied agreements

Target's PARALLEL PROTOCOL REQUEST response	Implied agreement
Non-zero REQ/ACK offset and a transfer period factor greater than 08h	Synchronous transfer (i.e., Each SCSI device transmits data with a period equal to or greater than and a REQ/ACK offset equal to or less than the negotiated values received in the target's PPR message).
Non-zero REQ/ACK offset and a transfer period factor equal 08h	Paced transfer (i.e., Each SCSI device transmits data with a period equal to the negotiated value and a REQ/ACK offset equal to or less than the negotiated values received in the target's PPR message).
REQ/ACK offset equal to zero	Asynchronous transfer
TRANSFER WIDTH EXPONENT equal to 1	16-bit data
TRANSFER WIDTH equal to zero	Eight-bit data
Protocol options equal to 0h and transfer period factor less than or equal to 9h	Eight-bit/asynchronous transfer with PROTOCOL OPTIONS field set to 0h
IU_REQ, DT_REQ, and QAS_REQ equal to zero	ST DATA IN and ST DATA OUT phases to transfer data
PCOMP_EN bit equal to zero	Precompensation to be disabled at originating SCSI device
PCOMP_EN bit equal to one	Precompensation to be enabled at originating SCSI device
DT_REQ equal to one	DT DATA IN and DT DATA OUT phases with data group transfers
IU_REQ, and DT_REQ equal to one	DT DATA IN and DT DATA OUT phases with information units
IU_REQ, DT_REQ, and QAS_REQ equal to one	DT DATA IN and DT DATA OUT phases with information units and use QAS for arbitration
RTI bit equal to zero	Training information is not saved at the target
RTI bit equal to one	Training information is saved at the target
WR_FLOW bit equal to zero	Write streaming flow control disabled
WR_FLOW bit equal to one	Write streaming flow control enabled
RD_STRM bit equal to zero	Read streaming disabled
RD_STRM bit equal to one	Read streaming enabled
MESSAGE REJECT message	The initiator shall set eight-bit/asynchronous transfer with protocol options field set to 0h
Parity error (on responding message)	Eight-bit/asynchronous transfer with PROTOCOL OPTIONS field set to 0h
Unexpected bus free (as a result of the responding message)	Eight-bit/asynchronous transfer with PROTOCOL OPTIONS field set to 0h
No response	Eight-bit/asynchronous transfer with protocol options field set to 0h

If there is an unrecoverable parity error on the initial PARALLEL PROTOCOL REQUEST message (see 10.12.2 and 10.12.4) the initiator shall retain its previous data transfer mode and protocol options. If there is an unexpected bus free on the initial PARALLEL PROTOCOL REQUEST message the initiator shall retain its previous data transfer mode and protocol options.

16.3.10.2 PARALLEL PROTOCOL REQUEST negotiation

If the initiator recognizes that PARALLEL PROTOCOL REQUEST negotiation is required, it creates an attention condition and sends a PARALLEL PROTOCOL REQUEST message to begin the negotiating process. After successfully completing the MESSAGE OUT phase, the target shall respond with a PARALLEL PROTOCOL REQUEST message or a MESSAGE REJECT message.

If an abnormal condition prevents the target from responding with a PARALLEL PROTOCOL REQUEST message or with a MESSAGE REJECT message then both SCSI devices shall use the eight-bit/asynchronous transfer mode with all the protocol options bits set zero to indicate ST DATA IN and ST DATA OUT phases between the two SCSI devices.

Following a target's responding PARALLEL PROTOCOL REQUEST message, an implied agreement for data transfers shall not be considered to exist until;

a) the initiator receives the last byte of the PARALLEL PROTOCOL REQUEST message and parity is valid; and
b) the target does not detect an attention condition on the last byte of the PARALLEL PROTOCOL REQUEST message.

If the initiator does not support the target’s responding PARALLEL PROTOCOL REQUEST message’s values the initiator shall create an attention condition and the first message shall be a MESSAGE REJECT message.

If during the PARALLEL PROTOCOL REQUEST message the initiator creates an attention condition and the first message of the MESSAGE OUT phase is either a MESSAGE PARITY ERROR or MESSAGE REJECT message the data transfers shall be considered to be negated by both SCSI devices. In this case, both SCSI devices shall use the eight-bit/asynchronous transfer mode with all the protocol options bits set zero to indicate ST DATA IN and ST DATA OUT phases for data transfers between the two SCSI devices.

16.3.11 QAS REQUEST

The QAS REQUEST message is sent from a target that has QAS enabled to begin a QAS phase (see 10.5.3).

16.3.12 RESTORE POINTERS

The RESTORE POINTERS message is sent from a target to direct the initiator to copy the most recently saved command, data, and status pointers for the task to the corresponding active pointers. The command and status pointers shall be restored to the beginning of the present command and status areas. The data pointer shall be restored to the value at the beginning of the data area in the absence of a SAVE DATA POINTER message or to the value at the point at which the last SAVE DATA POINTER message occurred for that task.

When information unit transfers are enabled there are implied restore pointers. For more information on this see 14.1 and 14.3.3.

16.3.13 SAVE DATA POINTER

The SAVE DATA POINTER message is sent from a target to direct the initiator to copy the current data pointer to the saved data pointer for the current task.

16.3.14 SYNCHRONOUS DATA TRANSFER REQUEST

16.3.14.1 SYNCHRONOUS DATA TRANSFER REQUEST message description

SYNCHRONOUS DATA TRANSFER REQUEST (SDTR) messages (see table 59) are used to negotiate a synchronous transfer agreement between two SCSI devices.

Table 59 - SYNCHRONOUS DATA TRANSFER message format

Bit / Byte	7	6	5	4	3	2	1	0
0	EXTENDED MESSAGE (01h)							
1	EXTENDED MESSAGE LENGTH (03h)							
2	SYNCHRONOUS DATA TRANSFER REQUEST (01h)							
3	TRANSFER PERIOD FACTOR							
4	REQ/ACK OFFSET							

The TRANSFER PERIOD FACTOR field is defined in table 60.

Table 60 - TRANSFER PERIOD FACTOR field

Code	Description
00h-09h	Reserved (note 1)
0Ah	transfer period equals 25 ns (note 2)
0Bh	transfer period equals 30,3 ns (note 2)
0Ch	transfer period equals 50 ns (note 3)
0Dh-18h	transfer period equals the transfer period factor * 4 (note 3)
19h-31h	transfer period equals the transfer period factor * 4 (note 4)
32h-FFh	transfer period equals the transfer period factor * 4 (note 5)

note:
1 - Faster timings may be allowed by future SCSI parallel interface standards.
2 - Fast-40 data transfer rates that have a period equal to 25 ns or 30,3 ns.
3 - Fast-20 data transfer rates that have a period of less than or equal to 96 ns and greater than or equal to 50 ns.
4 - Fast-10 data transfer rates that have a period of less than or equal to 196 ns and greater than or equal to 100 ns.
5 - Fast-5 data transfer rates that have a period of less than or equal to 1 020 ns and greater than or equal to 200 ns.

The REQ/ACK OFFSET is the maximum number of REQ assertions allowed to be outstanding before a corresponding ACK assertion is received at the target. The size of a data transfer may be 1 or 2 bytes depending on what values, if any, have been previously negotiated through an exchange of WIDE DATA TRANSFER REQUEST messages or PPR messages. The REQ/ACK OFFSET value is chosen to prevent overflow conditions in the SCSI device's reception buffer and offset counter. A REQ/ACK OFFSET value of zero shall indicate asynchronous transfer mode and that the TRANSFER PERIOD FACTOR field shall be ignored; a value of FFh shall indicate unlimited REQ/ACK offset.

An SDTR agreement applies to all logical units of the two SCSI devices that negotiated agreement. That is, if SCSI device A, acting as an initiator negotiates a synchronous transfer agreement with SCSI device B (a target), then the same data transfer agreement applies to SCSI devices A and B even if SCSI device B changes to an initiator.

A synchronous transfer agreement only applies to the two SCSI devices that negotiate the agreement. Separate synchronous transfer agreements are negotiated for each pair of SCSI devices. The synchronous transfer agreement only applies to DATA phases.

An SDTR message exchange shall be initiated by a target whenever a previously arranged synchronous transfer agreement may have become invalid. An SDTR message exchange shall be initiated by an initiator if that initiator does not support the PPR message or the initiator has determined the target does not support the PPR message whenever a previously arranged synchronous transfer agreement may have become invalid. The agreement becomes invalid after any condition that may leave the data transfer agreement in an indeterminate state such as:

a) after a hard reset;
b) after a TARGET RESET message;
c) after a power cycle; and
d) after a change in the transceiver mode (e.g., LVD mode to MSE mode).

Any condition that leaves the data transfer agreement in an indeterminate state shall cause the SCSI device to enter an asynchronous transfer mode and any protocol options shall no longer be in effect (i.e., DT DATA phase, paced transfers, information unit transfers, data group transfers, and QAS are disabled) (see 16.3.10).

A SCSI device may initiate an SDTR message exchange whenever it is appropriate to negotiate a new data transfer agreement (either synchronous or asynchronous). SCSI devices that are capable of synchronous transfers shall not respond to an SDTR message with a MESSAGE REJECT message.

Renegotiation after every selection is not recommended, since a significant performance impact is likely.

The SDTR message exchange establishes the permissible transfer periods and the REQ/ACK offsets for all logical units on the two SCSI devices. This agreement only applies to ST DATA IN phases and ST DATA OUT phases. COMMAND, MESSAGE, and STATUS phases shall use asynchronous transfers.

The originating SCSI device (the SCSI device that sends the first of the pair of SDTR messages) sets its values according to the rules above to permit it to receive data successfully. If the responding SCSI device is able to also receive data successfully with these values (or smaller transfer periods or larger REQ/ACK offsets or both), it returns the same values in its SDTR message. If it requires a larger transfer period, a smaller REQ/ACK offset, or both in order to receive data successfully, it substitutes values in its SDTR message as required, returning unchanged any value not required to be changed. Each SCSI device when transmitting data shall respect the negotiated limits set by the other's SDTR message, but it is permitted to transfer data with larger transfer periods, smaller synchronous REQ/ACK offsets, or both. The completion of an exchange of SDTR messages implies an agreement as shown in table 61.

Table 61 - SDTR messages implied agreements

Responding SCSI device SDTR response	Implied agreement
Non-zero REQ/ACK offset	Synchronous transfer (i.e., Each SCSI device transmits data with a transfer period equal to or greater than, and a REQ/ACK offset equal to or less than, the values received in the other device's SDTR message) with ST DATA IN and ST DATA OUT phases. Any protocol options shall no longer be in effect (see 16.3.10).
REQ/ACK offset equal to zero	Asynchronous transfer and any protocol options shall no longer be in effect (see 16.3.10).
MESSAGE REJECT message	The originating SCSI device shall set asynchronous transfer and any protocol options shall no longer be in effect (see 16.3.10).
Parity error (on responding message)	Asynchronous transfer and any protocol options shall no longer be in effect (see 16.3.10).
Unexpected bus free (as a result of the responding message)	Asynchronous transfer and any protocol options shall no longer be in effect (see 16.3.10).
No response	Asynchronous transfer and any protocol options shall no longer be in effect (see 16.3.10).

If there is an unrecoverable parity error on the initial SDTR message (see 10.12.2 and 10.12.4) the initiating SCSI device shall retain its previous data transfer mode. If there is an unexpected bus free on the initial SDTR message the initiating SCSI device shall retain its previous data transfer mode.

16.3.14.2 Target initiated SDTR negotiation

If the target recognizes that SDTR negotiation is required, it sends an SDTR message to the initiator.

The initiator shall create an attention condition on the last byte of the SDTR message from the target, and the initiator shall respond with its SDTR message, MESSAGE PARITY ERROR message, or with a MESSAGE REJECT message.

If an abnormal condition prevents the initiator from responding with a SDTR message or with a MESSAGE REJECT message then both SCSI devices shall return to asynchronous transfer mode and any protocol options shall no longer be in effect (see 16.3.10) for data transfers between the two SCSI devices.

Following an initiator 's responding SDTR message, an implied agreement for synchronous operation shall not be considered to exist until the target leaves MESSAGE OUT phase, indicating that the target has accepted the SDTR negotiation.

If the target does not support any of the initiator's responding SDTR message's values the target shall switch to a MESSAGE IN phase and the first message shall be a MESSAGE REJECT message. In this case the implied agreement shall be considered to be negated and both SCSI devices shall use the asynchronous transfer mode and any protocol options shall no longer be in effect (see 16.3.10) for data transfers between the two SCSI devices.

If a parity error occurs, the implied agreement shall be reinstated if a retransmission of a subsequent pair of messages is successfully accomplished. After a vendor-specific number of retry attempts (greater than zero), if the target continues to receive parity errors, it shall terminate the retry activity. This is done by the target causing an unexpected bus free. The initiator shall accept such action as aborting the SDTR

negotiation, and both SCSI devices shall go to asynchronous transfer mode and any protocol options shall no longer be in effect (see 16.3.10) for data transfers between the two SCSI devices.

16.3.14.3 Initiator initiated SDTR negotiation

If the initiator recognizes that SDTR negotiation is required, it creates an attention condition and sends a SDTR message to begin the negotiating process. After successfully completing the MESSAGE OUT phase, the target shall respond with the SDTR message or a MESSAGE REJECT message.

If an abnormal condition prevents the target from responding with a SDTR message or with a MESSAGE REJECT message then both SCSI devices shall go to asynchronous transfer mode and any protocol options shall no longer be in effect (see 16.3.10) for data transfers between the two SCSI devices.

Following a target's responding SDTR message, an implied agreement for synchronous transfers shall not be considered to exist until;

a) the initiator receives the last byte of the SDTR message and parity is valid; and
b) the target does not detect an attention condition on the last byte of the SDTR message.

If the initiator does not support the target's responding SDTR message's values the initiator shall create an attention condition and the first message shall be a MESSAGE REJECT message.

If during the SDTR message the initiator creates an attention condition and the first message out is either MESSAGE PARITY ERROR or MESSAGE REJECT the synchronous operation shall be considered to be negated by both the initiator and the target. In this case, both SCSI devices shall go to asynchronous transfer mode and any protocol options shall no longer be in effect (see 16.3.10) for data transfers between the two SCSI devices.

16.3.15 TASK COMPLETE

The TASK COMPLETE message is sent from a target to an initiator to indicate that a task has completed and that valid status has been sent to the initiator when information unit transfers are disabled.

After successfully sending this message the target shall go to the BUS FREE phase by releasing the BSY signal. The target shall consider the message transmission to be successful when there is no attention condition on the TASK COMPLETE message.

The task may have completed successfully or unsuccessfully as indicated in the status.

16.3.16 WIDE DATA TRANSFER REQUEST

16.3.16.1 WIDE DATA TRANSFER REQUEST message description

WIDE DATA TRANSFER REQUEST (WDTR) messages (see table 62) are used to negotiate a wide data transfer agreement between two SCSI devices.

Table 62 - WIDE DATA TRANSFER message format

Bit / Byte	7	6	5	4	3	2	1	0
0	EXTENDED MESSAGE (01h)							
1	EXTENDED MESSAGE LENGTH (02h)							
2	WIDE DATA TRANSFER REQUEST (03h)							
3	TRANSFER WIDTH EXPONENT (m)							

The TRANSFER WIDTH EXPONENT field defines the transfer width to be used during ST DATA IN phases and ST DATA OUT phases. The transfer width that is established applies to both SCSI devices. Valid transfer widths are 8 bits (m = 00h) and 16 bits (m = 01h). A TRANSFER WIDTH EXPONENT field value of 02h is obsolete and values greater than 02h are reserved.

A WDTR agreement applies to all logical units of the two SCSI devices that negotiated agreement. That is, if SCSI device A, acting as an initiator negotiates a wide data transfer agreement with SCSI device B (a target), then the same transfer width agreement applies to SCSI devices A and B even if SCSI device B changes to an initiator.

A wide data transfer agreement only applies to the two SCSI devices that negotiate the agreement. Separate wide transfer agreements are negotiated for each pair of SCSI devices. The wide data transfer agreement only applies to DATA phases.

A WDTR message exchange shall be initiated by a SCSI device whenever a previously arranged wide transfer agreement may have become invalid. The agreement becomes invalid after any condition that may leave the wide transfer agreement in an indeterminate state such as:

a) after a hard reset;
b) after a TARGET RESET message;
c) after a power cycle; and
d) after a change in the transceiver mode (e.g., LVD mode to MSE mode).

Any condition that leaves the data transfer agreement in an indeterminate state shall cause the SCSI device to enter an eight-bit wide data transfer mode.

A SCSI device may initiate a WDTR message exchange whenever it is appropriate to negotiate a new wide transfer agreement. SCSI devices that are capable of wide data transfers (greater than 8 bits) shall not respond to a WDTR message with a MESSAGE REJECT message.

Renegotiation after every selection is not recommended, since a significant performance impact is likely.

The WDTR message exchange establishes an agreement between the two SCSI devices on the width of the data path to be used for DATA phase transfers between two SCSI devices. This agreement only applies to ST DATA IN phases and ST DATA OUT phases. All other information transfer phases, except DT DATA phases, shall use an eight-bit data path.

If a SCSI device implements both wide data transfer option and synchronous transfer option and uses the SDTR and WDTR messages, then it shall negotiate the wide data transfer agreement prior to negotiating the synchronous transfer agreement. If a synchronous transfer agreement is in effect, then:

a) if either WDTR message is rejected with a MESSAGE REJECT message the prior synchronous

transfer agreement and any protocol option bits (see 16.3.10) shall remain intact;

b) If either WDTR message fails for any other reason the prior synchronous transfer agreement and any protocol option bits (see 16.3.10) shall remain intact; or

c) if both WDTR messages are not rejected with a MESSAGE REJECT message the WDTR message shall cause a reset of the synchronous transfer agreement to asynchronous mode and any protocol options shall no longer be in effect (i.e., DT DATA phase, paced transfers, information unit transfers, data group transfers, and QAS are disabled) (see 16.3.10).

The originating SCSI device (the SCSI device that sends the first of the pair of WDTR messages) sets its transfer width value to the maximum data path width it elects to accommodate. If the responding SCSI device is able to also accommodate this transfer width, it returns the same value in its WDTR message. If it requires a smaller transfer width, it substitutes the smaller value in its WDTR message. The successful completion of an exchange of WDTR messages implies an agreement as shown in table 63.

Table 63 - WDTR messages implied agreements

Responding SCSI device WDTR response	Implied agreement
TRANSFER WIDTH EXPONENT equal to 1	16-bit data
TRANSFER WIDTH equal to zero	Eight-bit data transfer
MESSAGE REJECT message (as a result of the responding message)	The originating SCSI device shall set eight-bit data transfer
Parity error (on responding message)	Eight-bit data transfer
Unexpected bus free (as a result of the responding message)	Eight-bit data transfer
No response	Eight-bit data transfer

If there is an unrecoverable parity error on the initial WDTR message (see 10.12.2 and 10.12.4) the initiating SCSI device shall retain its previous data transfer mode. If there is an unexpected bus free on the initial WDTR message the initiating SCSI device shall retain its previous data transfer mode.

16.3.16.2 Target initiated WDTR negotiation

If the target recognizes that WDTR negotiation is required, it sends a WDTR message to the initiator.

The initiator shall create an attention condition on the last byte of the WDTR message from the target, and the initiator shall respond with its WDTR message, MESSAGE PARITY ERROR message, or with a MESSAGE REJECT message.

If an abnormal condition prevents the initiator from responding with a WDTR message or with a MESSAGE REJECT message then both SCSI devices shall go to eight-bit data transfer mode for data transfers between the two SCSI devices.

Following an initiator's responding WDTR message, an implied agreement for wide data transfers operation shall not be considered to exist until the target leaves the MESSAGE OUT phase, indicating that the target has accepted the negotiation.

If the target does not support the initiator’s responding TRANSFER WIDTH EXPONENT the target shall switch to a MESSAGE IN phase and the first message shall be a MESSAGE REJECT message. In this case the implied agreement shall be considered to be negated and both SCSI devices shall use the eight-bit data transfer mode for data transfers between the two SCSI devices. Any prior synchronous transfer agreement shall remain intact.

If a parity error occurs, the implied agreement shall be reinstated if a retransmission of a subsequent pair of messages is successfully accomplished. After a vendor-specific number of retry attempts (greater than zero), if the target continues to receive parity errors, it shall terminate the retry activity. This is done by the target causing an unexpected bus free. The initiator shall accept such action as aborting the WDTR negotiation, and both SCSI devices shall go to eight-bit data transfer mode for data transfers between the two SCSI devices. Any prior synchronous transfer agreement shall remain intact.

16.3.16.3 Initiator initiated WDTR negotiation

If the initiator recognizes that WDTR negotiation is required, it creates an attention condition and sends a WDTR message to begin the negotiating process. After successfully completing the MESSAGE OUT phase, the target shall respond with a WDTR message or a MESSAGE REJECT message.

If an abnormal condition prevents the target from responding with a WDTR message or with a MESSGE REJECT message then both SCSI devices shall go to eight-bit transfer mode for data transfers between the two SCSI devices.

Following a target's responding WDTR message, an implied agreement for wide data transfers shall not be considered to exist until;

a) the initiator receives the last byte of the WDTR message and parity is valid; and
b) the target does not detect an attention condition before the ACK signal is released on the last byte of the WDTR message.

If the initiator does not support the target's responding TRANSFER WIDTH EXPONENT the initiator shall create an attention condition and the first message shall be a MESSAGE REJECT message.

If during the responding WDTR message the initiator creates an attention condition and the first message of the MESSAGE OUT phase is either a MESSAGE PARITY ERROR or MESSAGE REJECT message the wide data transfers shall be considered to be negated by both SCSI devices. In this case, both SCSI devices shall use the eight-bit data transfer mode for data transfers between the two devices.

16.4 Task attribute messages

16.4.1 Task attribute message overview and codes

Two byte task attribute messages are used to specify an identifier, called a tag, for a task that establishes the I_T_L_Q nexus. The TAG field is an 8-bit unsigned integer assigned by the application client and sent to the initiator in the send SCSI command request (see 19.3.2). The tag for every task for each I_T_L nexus shall be uniquely assigned by the application client. There is no requirement for the task manager to check whether a tag is currently in use for another I_T_L nexus. If the task manager checks the tag value and receives a tag that is currently in use for the I_T_L nexus, then it shall abort all tasks for the initiator and the associated logical unit and shall return CHECK CONDITION status for the task that caused the overlapped tag. The sense key shall be set to ABORTED COMMAND and the additional sense code shall be set to OVERLAPPED COMMANDS ATTEMPTED with the additional sense code qualifier set to the value of the duplicate tag (see 17.3). A tag becomes available for reassignment when the task ends. The numeric value of a tag is arbitrary, providing there are no outstanding duplicates, and shall not affect the order of execution.

For each logical unit on each target, each application client has up to 256 tags to assign to tasks. Thus a target with eight logical units could have up to 14 336 tasks concurrently in existence if there were seven initiators on the bus.

Whenever an initiator does a physical connection to a target, the appropriate task attribute message shall be sent following the IDENTIFY message to establish the I_T_L_Q nexus for the task. Only one I_T_L_Q nexus may be established during an initial connection or physical reconnection. If a task attribute message

is not sent, then only an I_T_L nexus is established for the task (i.e., an untagged command).

Whenever a target does a physical reconnection to an initiator to continue a tagged task, the SIMPLE QUEUE message shall be sent following the IDENTIFY message to resume the I_T_L_Q nexus for the task. Only one I_T_L_Q nexus may occur during a physical reconnection. If the SIMPLE TAG message is not sent, then only an I_T_L nexus occurs for the task (i.e., an untagged command).

If a target attempts to do a physical reconnection using an invalid tag, then the initiator should create an attention condition. After the corresponding MESSAGE OUT phase the initiator shall respond with an ABORT TASK message.

If a target does not implement tagged queuing and a queue tag message is received the target shall switch to a MESSAGE IN phase with a MESSAGE REJECT message and accept the task as if it were untagged provided there are no outstanding untagged tasks from that initiator.

See SCSI Architecture Model-2 standard for the task set management rules.

Table 64 - Task attribute message codes

Code	Support		Message Name	Direction		Clear Attention Condition
	Initiator	Target				
24h	O	O	ACA		Out	Not required
21h	Q	Q	HEAD OF QUEUE		Out	Not required
0Ah	O	O	LINKED COMMAND COMPLETE	In		n/a
22h	Q	Q	ORDERED		Out	Not required
20h	Q	Q	SIMPLE	In	Out	Not required

Key: M=Mandatory support, O=Optional support, Q=Mandatory if tagged queuing is implemented
In=Target to initiator, Out=Initiator to target
Yes=Initiator shall clear the attention condition before last ACK of the MESSAGE OUT phase.
Not required=Initiator may or may not clear the attention condition before last ACK of the MESSAGE OUT phase (see 12.2).
n/a=Not applicable
***=Extended message

16.4.2 ACA

See table 65 for the format of the ACA message.

Table 65 - ACA message format

Bit / Byte	7	6	5	4	3	2	1	0
0	MESSAGE CODE (24h)							
1	TAG (00h-FFh)							

The ACA message specifies that the task shall be placed in the task set as an ACA task. The rules used by

the task manager to handle ACA tasks within a task set are defined in the SCSI Architecture Model-2 standard.

16.4.3 HEAD OF QUEUE

See table 66 for the format of the HEAD OF QUEUE message.

Table 66 - HEAD OF QUEUE message format

Bit / Byte	7	6	5	4	3	2	1	0
0	MESSAGE CODE (21h)							
1	TAG (00h-FFh)							

The HEAD OF QUEUE message specifies that the task shall be placed in the task set as a HEAD OF QUEUE task. The rules used by the device server to handle HEAD OF QUEUE tasks within a task set are defined in the SCSI Architecture Model-2 standard.

16.4.4 LINKED COMMAND COMPLETE

The LINKED COMMAND COMPLETE message is sent from a target to an initiator to indicate that a linked command has completed and that status has been sent. The initiator shall then set the pointers to the initial state for the next linked command.

16.4.5 ORDERED

See table 67 for the format of the ORDERED message.

Table 67 - ORDERED message format

Bit / Byte	7	6	5	4	3	2	1	0
0	MESSAGE CODE (22h)							
1	TAG (00h-FFh)							

The ORDERED message specifies that the task shall be placed in the task set as an ORDERED task. The rules used by the task manager to handle ORDERED tasks within a task set are defined in the SCSI Architecture Model-2 standard.

16.4.6 SIMPLE

See table 68 for the format of the SIMPLE message.

Table 68 - SIMPLE message format

Bit / Byte	7	6	5	4	3	2	1	0
0	MESSAGE CODE (20h)							
1	TAG (00h-FFh)							

The SIMPLE message specifies that the task shall be placed in the task set as a SIMPLE task. The rules used by the task manager to handle SIMPLE tasks within a task set are defined in the SCSI Architecture Model-2 standard.

16.5 Task management messages

16.5.1 Task management message codes

Table 69 - Task management message codes

Code	Support		Message Name	Direction		Clear Attention Condition
	Initiator	Target				
0Dh	Q	Q	ABORT TASK		Out	Yes
06h	O	M	ABORT TASK SET		Out	Yes
16h	O	O	CLEAR ACA		Out	Not required
0Eh	Q	Q	CLEAR TASK SET		Out	Yes
17h	O	O	LOGICAL UNIT RESET (Note)		Out	Yes
0Ch	O	M	TARGET RESET		Out	Yes

Key: M=Mandatory support, O=Optional support, Q=Mandatory if tagged queuing is implemented
In=Target to initiator, Out=Initiator to target
Yes=Initiator shall clear the attention condition before last ACK of the MESSAGE OUT phase.
Not required=Initiator may or may not clear the attention condition before last ACK of the MESSAGE OUT phase (see 12.2).
n/a=Not applicable
***=Extended message

Note-The LOGICAL UNIT RESET message is mandatory if hierarchical addressing (see SCSI Architecture Model-2 standard) is implemented by the target.

16.5.2 ABORT TASK

The ABORT TASK message is defined in the SCSI Architecture Model-2 standard.

In addition to the requirements in the SCSI Architecture Model-2 standard the target shall go to the BUS FREE phase following the successful receipt of the ABORT TASK message.

If only an I_T nexus has been established, the target shall go to the BUS FREE phase. No status or message shall be sent for the current task and no pending data, status, or tasks are affected.

NOTE 44 - The ABORT TASK message in the case of only an I_T nexus is useful to an initiator that is not able to get an IDENTIFY message through to the target due to parity errors and just needs to end the current task. Any pending data, status, or tasks for the I_T nexus are not affected. It is not possible to abort an I_T nexus on a physical reconnection because of item (f) in 12.2.

On a physical reconnection, the ABORT TASK message aborts the current task if it is fully identified. If the current task is not fully identified (i.e. an I_T_L nexus exists, but the target is doing a physical reconnecting for an I_T_L_Q nexus), then the current task is not aborted and the target goes to the BUS FREE phase.

NOTE 45 - A nexus may not be fully identified on a physical reconnection if an attention condition is created during the IDENTIFY message and the target has any tagged tasks for that initiator on that logical unit.

It is not an error to issue this message to an I_T_L or I_T_L_Q nexus that does not have any pending tasks.

16.5.3 ABORT TASK SET

The ABORT TASK SET message is defined in the SCSI Architecture Model-2 standard.

In addition to the requirements in the SCSI Architecture Model-2 standard the target shall go to the BUS FREE phase following the successful receipt of the ABORT TASK SET message.

If only an I_T nexus has been established, the target shall switch to a BUS FREE phase. No status or message shall be sent for the current task and no pending data, status, or tasks are affected.

The ABORT TASK SET message in the case of only an I_T nexus is useful to an initiator that is not able to send an IDENTIFY message through to the target due to parity errors and just needs to end the current task or task management function.

It is not an error to issue this message to an I_T_L nexus that does not have any pending or current tasks.

16.5.4 CLEAR ACA

The CLEAR ACA message is defined in the SCSI Architecture Model-2 standard.

The CLEAR ACA message shall only be sent by an initiator during an initial connection. If the target receives the CLEAR ACA message at any other time the target shall switch to a MESSAGE IN phase and issue a MESSAGE REJECT message. The target shall then continue processing the task that was in process when the CLEAR ACA message was received.

On receipt of a CLEAR ACA message the task manager, in addition to clearing the ACA condition, shall go to the BUS FREE phase following the successful receipt of the CLEAR ACA message.

It is not an error to issue a CLEAR ACA message when no ACA condition is in effect.

16.5.5 CLEAR TASK SET

The CLEAR TASK SET message is defined in the SCSI Architecture Model-2 standard.

In addition to the requirements in the SCSI Architecture Model-2 standard the target shall go to the BUS FREE phase following the successful receipt of the CLEAR TASK SET message.

16.5.6 LOGICAL UNIT RESET

The LOGICAL UNIT RESET message is defined in the SCSI Architecture Model-2 standard.

If only an I_T nexus has been established the LOGICAL UNIT RESET shall be performed as if it were a TARGET RESET.

In addition to the requirements in the SCSI Architecture Model-2 standard the target shall go to the BUS FREE phase following the successful receipt of the LOGICAL UNIT RESET message.

16.5.7 TARGET RESET

The TARGET RESET message is defined in the SCSI Architecture Model-2 standard.

In addition to the requirements in the SCSI Architecture Model-2 standard the target, following the successful receipt of the TARGET RESET message shall go to the BUS FREE phase.

17 Command processing considerations and exception conditions

17.1 Command processing considerations and exception conditions overview

The following subclauses describe some aspects of command processing, including exception conditions and error handling that are specific to this standard.

17.2 Asynchronous event notification

Notification of an asynchronous event is performed using the SEND command with the AER bit set to one. The information identifying the condition being reported shall be returned during the data out delivery phase of the SEND command (see SCSI Primary Commands-2 standard).

An error condition or unit attention condition shall be reported once per occurrence of the event causing it. The target may choose to use an asynchronous event notification or to return CHECK CONDITION status on a subsequent command, but not both. Notification of command-related error conditions shall be sent only to the initiator that requested the task.

The asynchronous event notification protocol may be used to notify processor devices that a system resource has become available. If a target chooses to use this method, the sense key in the sense data sent to the processor device shall be set to UNIT ATTENTION.

The asynchronous event notification protocol shall be used only with SCSI devices that return processor device type with an AERC bit of one in response to an INQUIRY command. The INQUIRY command should be sent to logical unit zero of each SCSI device responding to selection. This procedure shall be conducted prior to the first asynchronous event notification and shall be repeated whenever the SCSI device deems it appropriate or when an event occurs that may invalidate the current information. (See SYNCHRONOUS DATA TRANSFER REQUEST message (16.3.14) for examples of these events.)

Each SCSI device that returns processor device type with an AERC bit of one shall be issued a TEST UNIT READY command to determine that the SCSI device is ready to receive an asynchronous event notification. A SCSI device returning CHECK CONDITION status is issued a REQUEST SENSE command. This clears any pending unit attention condition. A SCSI device that returns processor device type with an AERC bit of one and returns GOOD status when issued a TEST UNIT READY command shall accept a SEND command with an AER bit of one.

> NOTE 46 - A SCSI device that uses asynchronous event notification at initialization time should provide means to defeat these notifications. This may be done with a switch or jumper wire. SCSI devices that implement saved parameters may alternatively save the asynchronous event notification permissions either on a per SCSI device basis or as a system wide option. In any case, a SCSI device conducts a survey with INQUIRY commands to be sure that the SCSI devices on the SCSI bus are appropriate destinations for SEND commands with an AER bit of one. (The SCSI devices on the bus or the SCSI ID assignments may have changed.)

See asynchronous event reporting in the SCSI Architecture Model-2 standard for more information on asynchronous event notification.

17.3 Incorrect initiator connection

An incorrect initiator connection occurs during an initial connection if an initiator creates a nexus that already exists and does not send an ABORT TASK SET, ABORT TASK, TARGET RESET, CLEAR TASK SET, DISCONNECT, or LOGICAL UNIT RESET message as one of the messages of the MESSAGE OUT phase or as one of the task management flags in the SPI command information unit.

A task manager that detects an incorrect initiator connection shall abort all tasks for the initiator and the associated logical unit and shall return CHECK CONDITION status for the task that caused the incorrect

initiator connection. The sense key shall be set to ABORTED COMMAND and the additional sense code shall be set to OVERLAPPED COMMANDS ATTEMPTED with the additional sense code qualifier set to the value of the duplicate tag (see 16.4).

NOTE 47 - An incorrect initiator connection may be indicative of a serious error and, if not detected, could result in a task operating with a wrong set of pointers. This is considered a catastrophic failure on the part of the initiator. Therefore, vendor-specific error recovery procedures may be required to guarantee the data integrity on the medium. The target may return additional sense data to aid in this error recovery procedure (e.g., sequential-access devices may return the residue of blocks remaining to be written or read at the time the second command was received).

17.4 Unexpected RESELECTION phase

An unexpected RESELECTION phase occurs if a target attempts to do a physical reconnect to a task for which a nexus does not exist. An initiator should respond to an unexpected RESELECTION phase by sending an ABORT TASK message.

18 SCSI management features for the SCSI parallel interface

18.1 SCSI mode parameters

18.1.1 SCSI mode parameter overview and codes

This subclause describes the block descriptors and the pages used with MODE SELECT and MODE SENSE commands that influence, control and report the behavior of the SCSI parallel interface. All mode parameters not defined in this standard shall influence the behavior of the SCSI devices as specified in the appropriate command set document. The mode pages are addressed to the device server of a logical unit. The mode pages associated with the SCSI parallel interface are listed in table 70.

Table 70 - Mode page codes for the SCSI parallel interface

Page code	Description	Subclause
02h	Disconnect-reconnect page	18.1.2
18h	Logical Unit Control page	18.1.3
19h	Port Control page	18.1.4

18.1.2 Disconnect-reconnect mode page

The disconnect-reconnect page (see table 71) provides the application client the means to tune the performance of the SCSI parallel interface. The following subclause defines the fields in the disconnect-reconnect mode page of the MODE SENSE or MODE SELECT command that are used by targets.

The application client passes the fields used to control the SCSI parallel interface to a device server by means of a MODE SELECT command. The device server then communicates the field values to the target. The field values are communicated from the device server to the target in a vendor specific manner.

SPI SCSI devices shall only use disconnect-reconnect page parameter fields defined below. If any other fields within the disconnect-reconnect page of the MODE SELECT command contain a non-zero value, the device server shall return CHECK CONDITION status for that MODE SELECT command. The sense key shall be set to ILLEGAL REQUEST and the additional sense code set to ILLEGAL FIELD IN PARAMETER LIST.

Table 71 - Disconnect-reconnect page (02h)

Bit / Byte	7	6	5	4	3	2	1	0
0	PS	RESERVED	PAGE CODE (02h)					
1	PAGE LENGTH (0Eh)							
2	BUFFER FULL RATIO							
3	BUFFER EMPTY RATIO							
4	(MSB)	BUS INACTIVITY LIMIT						
5								(LSB)
6	(MSB)	PHYSICAL DISCONNECT TIME LIMIT						
7								(LSB)
8	(MSB)	CONNECT TIME LIMIT						
9								(LSB)
10	(MSB)	MAXIMUM BURST SIZE						
11								(LSB)
12	EMDP	FAIR ARBITRATION			DIMM	DTDC		
13	RESERVED							
14	RESERVED							
15								

The BUFFER FULL RATIO field and BUFFER EMPTY RATIO FIELD are used as described in the SCSI-3 Primary Commands standard.

The BUS INACTIVITY LIMIT field indicates the maximum time in 100 µs increments that the target is permitted to assert the BSY signal without a REQ/ACK handshake. If the bus inactivity limit is exceeded the target shall attempt to do a physical disconnect (see 16.3.2) if the initiator has granted the physical disconnect privilege (see 16.3.3) and it is not restricted by DTDC. This value may be rounded as defined in the SCSI Primary Commands-2 standard. A value of zero indicates that there is no bus inactivity limit.

The PHYSICAL DISCONNECT TIME LIMIT field indicates the minimum time in 100 µs increments that the target shall wait after releasing the SCSI bus before attempting a physical reconnection. This value may be rounded as defined in the SCSI Primary Commands-2 standard. A value of zero indicates that there is no physical disconnect time limit.

The CONNECT TIME LIMIT field indicates the maximum time in 100 µs increments that the target is allowed to use the SCSI bus before doing a physical disconnect, if the initiator has granted the physical disconnect privilege (see 16.3.3) and it is not restricted by DTDC. This value may be rounded as defined in the SCSI Primary Commands-2 standard. A value of zero indicates that there is no connect time limit.

If information unit transfers are disabled the MAXIMUM BURST SIZE field indicates the maximum amount of

data that the target shall transfer during a DATA phase before doing a physical disconnect if the initiator has granted the physical disconnect privilege (see 16.3.3).

If information unit transfer are enabled the MAXIMUM BURST SIZE field indicates the maximum amount of data that the target shall transfer in a single SPI data information unit.

The maximum burst size is expressed in increments of 512 bytes (e.g. a value of one means 512 bytes, two means 1 024 bytes, etc.). A value of zero indicates there is no limit on the amount of data transferred per burst.

The enable modify data pointer (EMDP) bit indicates whether or not the initiator allows the MODIFY DATA POINTER message to be issued by the target. If the EMDP bit is a zero, the target shall not issue the MODIFY DATA POINTER message. If the EMDP bit is a one, the target is allowed to issue MODIFY DATA POINTER messages.

If the EMDP bit is a one and the initiator responds to a MODIFY DATA POINTER message with a MESSAGE REJECT, then the target shall return a CHECK CONDITION. The sense key shall be set to ABORTED COMMAND and the sense code shall be set to INVALID MESSAGE ERROR.

If the FAIR ARBITRATION field is set to 000b, the target shall not use arbitration fairness during normal arbitration. If this field is set to a nonzero value, the target shall use arbitration fairness during normal arbitration (see Annex B).

Regardless of the value in the FAIR ARBITRATION field the target shall use arbitration fairness during QAS.

A disconnect immediate (DIMM) bit of zero indicates that the target may request DATA IN or DATA OUT phases following a COMMAND phase without attempting a physical disconnect (see 16.3.2). A DIMM bit of one indicates that the target shall attempt a physical disconnect (see 16.3.2) after a COMMAND phase and before a subsequent DATA IN or DATA OUT phase. The DIMM bit only applies when the initiator has granted the physical disconnect privilege (see 16.3.3).

The DATA TRANSFER DISCONNECT CONTROL (DTDC) field (see table 72) defines further restrictions on when a physical disconnect is permitted.

TABLE 72 - DATA TRANSFER DISCONNECT CONTROL

DTDC	Description
000b	DATA TRANSFER DISCONNECT CONTROL is not used. Physical disconnect is controlled by the other fields in this page.
001b	A target shall not attempt to do a physical disconnect once the data transfer of a command has started until all data the command is to transfer has been transferred. The connect time limit and bus inactivity limit are ignored during the data transfer.
010b	Reserved
011b	A target shall not attempt to do a physical disconnect once the data transfer of a command has started, until the command is complete. The connect time limit and bus inactivity limit are ignored once data transfer has started.
100b-111b	Reserved

If DTDC is non-zero and the maximum burst size is non-zero, the target shall return a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code set to ILLEGAL FIELD IN PARAMETER LIST.

18.1.3 Logical Unit Control mode page

The Logical Unit Control mode page (see table 73) contains those parameters that select logical unit operation options. This page is not currently defined for SPI SCSI devices. The implementation of any parameter and its associated functions is optional. The page follows the MODE SENSE / MODE SELECT rules specified by the SCSI Primary Commands-2 standard.

Table 73 - Logical Unit Control page (18h)

Bit / Byte	7	6	5	4	3	2	1	0
0	PS	RESERVED	PAGE CODE (18h)					
1	PAGE LENGTH (06h)							
2	RESERVED							
3	RESERVED							
4	RESERVED							
5	RESERVED							
6	RESERVED							
7	RESERVED							

18.1.4 Port Control mode page

The Port Control mode page (see table 74) contains those parameters that select SPI SCSI device port operation options. The page shall be implemented by LUN 0 of all SPI SCSI devices. The page shall not be implemented by logical units other than LUN 0. The implementation of any bit and its associated functions is optional. The page follows the MODE SENSE / MODE SELECT rules specified by SCSI Primary Commands-2 standard.

Table 74 - Port Control page (19h)

Bit / Byte	7	6	5	4	3	2	1	0
0	PS	RESERVED	PAGE CODE (19h)					
1	PAGE LENGTH (06h)							
2	RESERVED							
3	RESERVED				PROTOCOL IDENTIFIER (1h)			
4	(MSB) SYNCHRONOUS TRANSFER TIMEOUT							
5	(LSB)							
6	RESERVED							
7	RESERVED							

The PROTOCOL IDENTIFIER field indicates the protocol that this mode page applies to. The protocol identifier field has a value of 1h to indicate SPI SCSI devices.

The SYNCHRONOUS TRANSFER TIMEOUT field indicates the maximum amount of time in 1 millisecond increments that the target shall wait before generating an error by doing an unexpected bus free (see 10.3). The target shall only go to a BUS FREE phase if one of the following events causes the timer, once started, to not reset or reload before expiring.

a) If there is a REQ transition when there are no outstanding REQs waiting for an ACK then load and start the timer.
b) If there is a REQ transition when there are any outstanding REQs waiting for an ACK then there is no effect on the timer.
c) If there is an ACK transition when there are outstanding REQs waiting for an ACK then load and start the timer.
d) If after an ACK transition there are no outstanding REQs waiting for an ACK then stop the timer.

A SYNCHRONOUS TRANSFER TIMEOUT field value of 0000h indicates that the function is disabled. A value of FFFFh indicates an unlimited period.

19 SCSI parallel interface services

19.1 SCSI parallel interface services overview

SCSI parallel interface services are provided by the initiator enabling the application client to accomplish tasks and task management functions (see SCSI Architecture Model-2 standard) and by the target enabling the device server to receive commands and move data to/from an application client. The SCSI parallel interface services are described in terms of the services the initiator and target provide. Each SCSI parallel interface service causes a sequence phases to be generated by the addressed SCSI devices. Figure 61, figure 62, and figure 63 show all the valid phase sequences.

19.2 Procedure objects

See table 75 for the mapping of the procedure objects used in this standard to the equivalent procedure objects used in the SCSI Architecture Model-2 standard.

Table 75 - This standards objects mapped to objects from other SCSI standards

Objects	Equivalent SCSI Architecture Model-2 standard objects
initiator SCSI ID	initiator identifier
target SCSI ID	target identifier
initiator SCSI ID+target SCSI ID+logical unit number[+tag]	task identifier
target SCSI ID+logical unit number[+tag]	task address
target SCSI ID **or** target SCSI ID+logical unit number **or** initiator SCSI ID+target SCSI ID+logical unit number+tag	object identifier
target SCSI ID **or** target SCSI ID+logical unit number **or** target SCSI ID+logical unit number+tag	object address
target SCSI ID+initator SCSI ID	none

See table 76 for a list of the procedure objects used when passing services across the SCSI parallel interface service interface. See table 76 for the definitions of the names used within this standard and the equivalent SCSI Architecture Model-2 standard names of the procedure objects, the name of the standard where the objects are defined, the standard where the binary contents of the objects are defined, and the routing of the objects. The routing shows:

a) the originating object of the term,
b) the object that is the final destination of the term, and
c) the objects that the term moves though to reach the final destination object.

.

Table 76 - Procedure objects

This standard's objects	Standard where term defined	Standard where binary contents of term defined	Term routing
application client buffer offset	SAM-2	SAM-2	DS → targ→ init
command byte count	SAM-2	SAM-2	AC → init
command descriptor block	SAM-2	SAM-2/cmd (note 2)	AC → init → targ → DS
data-in buffer	SAM-2	cmd (note 3)	DS → targ→ init → AC
data-out buffer	SAM-2	cmd (note 3)	AC → init → targ → DS
device server buffer	SAM-2	cmd (note 3)	DS → targ→ init
initiator SCSI ID	SAM-2	this standard	DS → targ **or** TM → targ
link control function	this standard	this standard	AC → init → targ
logical unit number	SAM-2	this standard	AC → init → targ → DS **or** AC → init → targ → TM **or** DS → targ→ init
request byte count	SAM-2	SAM-2	DS → targ
service response	SAM-2	this standard (note 4)	DS → targ→ init → AC **or** targ → DS
service response (note 1)	SAM-2	this standard (note 4)	init → AC
status	SAM-2	SAM-2	DS → targ→ init → AC
tag	SAM-2	this standard	AC → init → targ → DS **or** AC → init → targ → TM **or** DS → targ→ init
target SCSI ID	SAM-2	this standard	AC → init → targ → DS **or** AC → init → targ → TM **or** DS → targ
target SCSI ID+initator SCSI ID	this standard	this standard	targ → DS **or** targ → TM
task attribute	SAM-2	this standard	AC → init → targ → DS

Key: AC=application client, cmd=SCSI command standards, DS=device server, init=initiator, SAM-2=SCSI Architecture Model-2 standard, TM=task manager, targ=target

Notes

1) Only occurs when unexpected bus free (see 10.3) is detected by the initiator.
2) The portions not defined in the SCSI Architecture Model-2 standard are defined in the SCSI command standards (e.g.,SCSI-3 Block Commands standard, SCSI Primary Commands-2 standard).
3) Parameter lists are defined within one of the SCSI command standards (e.g.,SCSI-3 Block Commands standard, SCSI Primary Commands-2 standard). SCSI standards do not define non-parameter list information.
4) The SERVICE DELIVERY OR TARGET FAILURE value of the service response is not defined in SCSI.

19.3 Application client SCSI command services

19.3.1 Application client SCSI command services overview

The SCSI command services shall be requested by the application client using a procedure call defined as:

Service response = execute command (target SCSI ID+logical unit number[+tag], command descriptor block, [task attribute], [link control function], [data-out buffer], [command byte count] ||| [data-in buffer], status, service response).

19.3.2 Send SCSI command service

The send SCSI command service is a four step confirmed service that provides the means to transfer a command data block to a device server.

Processing the execute command procedure call for a send SCSI command service shall be composed of the 4 step confirmed service shown in table 77.

Table 77 - Processing of send SCSI command service procedure

Step	Protocol service name	SCSI Protocol Service Interface procedure call
request	send SCSI command request	send SCSI command (target SCSI ID+logical unit number[+tag], command descriptor block, [task attribute], [link control function], [data-out buffer], [command byte count] ||).
indication	send SCSI command indication	SCSI command received (target SCSI ID+initator SCSI ID+logical unit number[+tag], [command descriptor block], [task attribute], ||).
response	send SCSI command response	send command complete (target SCSI ID+initator SCSI ID+logical unit number[+tag], [status], [service response], ||).
confirmation	send SCSI command confirmation	command complete received (target SCSI ID+logical unit number[+tag], [data-in buffer], [status], service response ||).

19.4 Device server SCSI command services

19.4.1 Device server SCSI command services overview

The SCSI data buffer movement services shall be requested from the device server using a procedure call defined as:

Service response = move data buffer (target SCSI ID+initator SCSI ID+logical unit number [+tag], device server buffer, application client buffer offset, request byte count ||).

Only one type of data buffer movement procedure call shall be used while processing one command, either data-in delivery or data-out delivery.

19.4.2 Data-in delivery service

The data-in delivery service is a two step confirmed service that provides the means to transfer a parameter list or data from a device server to an initiator.

Processing the execute command procedure call for a data-in delivery service shall be composed of the 2 step confirmed service shown in table 78.

Table 78 - Processing of data-in delivery service procedure

Step	Protocol service name	SCSI Protocol Service Interface procedure call
request	data-in delivery request	send data-in (target SCSI ID+initator SCSI ID+logical unit number[+tag], device server buffer, application client buffer offset, request byte count ‖).
confirmation	data-in delivery confirmation	data delivered (target SCSI ID+initator SCSI ID+logical unit number[+tag], service response ‖).

19.4.3 Data-out delivery service

The data-out delivery service is a two step confirmed service that provides the means to transfer a parameter list or data from an initiator to a device server.

Processing the execute command procedure call for a data-out delivery service shall be composed of the 2 step confirmed service shown in table 79.

Table 79 - Processing of data-out delivery service procedure

Step	Protocol service name	SCSI Protocol Service Interface procedure call
request	data-out delivery request	receive data-out (target SCSI ID+initator SCSI ID+logical unit number[+tag], application client buffer offset, request byte count, device server buffer ‖).
confirmation	data-out delivery confirmation	data-out received (target SCSI ID+initator SCSI ID+logical unit number [+tag] service response ‖)

19.5 Task management services

19.5.1 Task management services overview

The task management services shall be requested from the application client using a procedure call defined as:

Service response = task management function (target SCSI ID|target SCSI ID+logical unit number|target SCSI ID+logical unit number+tag, service delivery failure flag ‖ service response).

19.5.2 Task management function service

This standard handles task management functions as a four step confirmed service that provides the means to transfer task management functions to a task manager.

The task management functions are defined in the SCSI Architecture Model-2 standard. This standard defines the actions taken by the SCSI parallel interface service to carry out the requested task management functions.

19.5.3 ABORT TASK

The SCSI parallel interface services request the initiator issue an ABORT TASK message (see 16.5.2) to the selected SCSI device.

19.5.4 ABORT TASK SET

The SCSI parallel interface services request the initiator issue an ABORT TASK SET message (see 16.5.3) to the selected SCSI device.

19.5.5 CLEAR ACA

The SCSI parallel interface services request the initiator issue a CLEAR ACA message (see 16.5.4) to the selected SCSI device.

19.5.6 CLEAR TASK SET

The SCSI parallel interface services request the initiator issue a CLEAR TASK SET message (see 16.5.5) to the selected SCSI device.

19.5.7 LOGICAL UNIT RESET

The SCSI parallel interface services request the initiator issue a LOGICAL UNIT REST message (see 16.5.6) to the selected SCSI device.

19.5.8 RESET SERVICE DELIVERY SUBSYSTEM

The SCSI parallel interface services request the initiator issue a hard reset (see 12.3) to the selected SCSI device.

19.5.9 TARGET RESET

The SCSI parallel interface services request the initiator issue a TARGET RESET message (see 16.5.7) to the selected SCSI device.

19.5.10 WAKEUP

The SCSI parallel interface services request the initiator issue a hard reset (see 12.3) to the selected SCSI device.

Annex A

(normative)

Additional requirements for LVD SCSI drivers and receivers

A.1 System level requirements

The requirements for LVD SCSI drivers and receivers in this annex are based on the system level requirements stated in table A.1. Some of these requirements are specifically called out in other subclauses while others are derived from bus loading conditions and trade-offs between competing parameters.

Table A.1 - System level requirements

Parameter	Minimum	Maximum	Cross-reference
V_A (paced transfers non-isolated transition) (except OR-tied signals)	-1 V	-100 mV	note 1, 5
V_A (paced transfers isolated noncompensated transition) (except OR-tied signals)	-1 V	50 mV	note 6
V_A (paced transfers isolated compensated transition) (except OR-tied signals)	-1 V	TBD mV	
V_N (paced transfers non-isolated transition) (except OR-tied signals)	100 mV	1 V	note 1, 5
V_N (paced transfers isolated noncompensated transition) (except OR-tied signals)	-50 mV	1 V	note 6
V_N (paced transfers isolated compensated transition) (except OR-tied signals)	TBD mV	1 V	
V_A (OR-tied signals)	-3,6 V	-100 mV	note 1
V_N (OR-tied signals)	80 mV	145 mV	note 1, 8
attenuation (%) (fast-40 and fast-80)		15	note 2
attenuation (%) (fast-160)		50%	note 7
loaded media impedance (Ohms)	85	135	note 3
unloaded media impedance (Ohms)	110	135	subclause 6.3
terminator bias (mV)	100	125	subclause 7.3.1
terminator impedance (Ohms)	100	110	subclause 7.3.1
device leakage (µA)	-20	20	table 16
number of SCSI devices	2	16	subclause 4.7
ground offset level (mV)	-355	355	note 4

Note:
1 -These are the signal levels at the receiver, the system allows 60 mV crosstalk for calculating the minimum driver level.
2 -Measured from the driver to the farthest receiver.
3 -Caused by the addition of device capacitive load (see table 9 for calculations).
4 -This is the difference in voltage signal commons for SCSI devices on the bus (see figure 3).
5 -Clock and non-isolated transitions: toggling signal after the first transition must be the 100 mV limits specified by the fast-80 or slower speeds. These values only apply to isolated transitions.
6 - Weak driver is disabled
7 - Attenuation is the sum of the DC Loss 10% and AC loss 40% at 80 MHz.
8 - Prior versions of the standard did not account for leakage.

A.2 Driver requirements

A.2.1 Driver requirements overview

The fundamental requirement for an LVD driver is the generation of a first-step differential output voltage magnitude at the driver connections to the balanced media to achieve required minimum differential signals at every receiver connection to the bus. If ~~a P_EN bit of one was received by a SCSI device during the prior PPR negotiation~~precompensation is enabled, the weak driver amplitude shall minimum of ~~60~~ 50 % to a maximum of ~~78~~ 66 % of the strong driver amplitude after the first bit of a series of adjacent ones or adjacent zeros. Other characteristics that affect overall noise margin are the common-mode output voltage, the maximum differential output voltage, the driver output impedance, and the output signal wave shape.

The driver requirements are defined in terms of the voltages and currents depicted in figure 44.

A.2.2 Differential output voltage, V_S

This subclause does not specify requirements for drivers with source impedances less then 1000 Ohms.

To assure sufficient voltage to define a valid logic state at any device connection on a fully loaded LVD bus at least a minimum differential output voltage shall be generated. This value shall be large enough that, after allowance for attenuation (AC and DC), reflections, terminator bias difference, and differential noise coupling, V_S is at least ±100 mV at the device connector to the LVD SCSI bus.

The SCSI device shall also comply with the upper limits for the differential output voltages and to the symmetry of the differential output voltage magnitudes between logic states in order to assure a first-step transition to the opposite logic state~~.~~.

During paced transfers if precompensation is enabled the signal level at the receiver shall be a minimum of 50 mV V_S on an isolated assertion or negation. During paced transfers if precompensation is disabled the signal level at the receiver shall be a minimum of –35 mV V_S on an isolated assertion or negation.

~~With~~ For synchronous transfers, with the test circuit of figure A.1 and the test conditions V1 and V2 in table A.2 applied, the steady-state magnitude of the differential output voltage, V_S, for an asserted state (V_A), shall be greater than or equal to 320 mV and less than or equal to 800 mV. For the negated state, the polarity of V_S shall be reversed (V_N) and the differential voltage magnitude shall be greater than or equal to 320 mV and less than or equal to 800 mV. The relationship between V_A and V_N specified in table A.2 and shown graphically in figure A.2 shall be maintained.

For paced transfers, with the test circuit of figure A.1 and the test conditions V1 and V2 in table A.3 applied, the steady-state magnitude of the differential output voltage, V_S, for an asserted state (V_A), shall be greater than or equal to 370 mV and less than or equal to 800 mV. For the negated state, the polarity of V_S shall be reversed (V_N) and the differential voltage magnitude shall be greater than or equal to 370 mV and less than or equal to 800 mV. The relationship between V_A and V_N specified in table A.3 and shown graphically in figure A.3 shall be maintained. If precompensation is enabled the signal level for the strong driver shall be capable of greater than or equal to 500 mV and shall be less than or equal to 800 mV in the V_N and V_A states. The strong driver relationship between V_A and V_N specified in table A.2 strong driver and shown graphically in figure A.3 for the strong shall be maintained.

The assertion drivers and negation drivers require different strengths to achieve the near equality in V_A and V_N shown in figure A.2 and figure A.3 because the applied V1 and V2 simulate the effects of the bus termination bias.

Table A.2 - Driver steady-state test limits and conditions for synchronous transfers

Test parameter	Test conditions (figure A.1)(note 1)	Minimum (mV) (note 2)	Maximum (mV)
$\lvert V_A \rvert$ Differential output voltage magnitude (asserted) (note)	V_1 = 1,056 V V_2 = 0,634 V	320	800
	V_1 = 1,866 V V_2 = 1,444V	320	800
$\lvert V_N \rvert$ Differential output voltage magnitude (negated) (note)	V_1 = 1,056 V V_2 = 0,634 V	320	800
	V_1 = 1,866 V V_2 = 1,444V	320	800
$\lvert V_A \rvert$ Differential output voltage magnitude (asserted)		0,69 x $\lvert V_N \rvert$ + 50	1,45 x $\lvert V_N \rvert$ - 65

Notes:
1 The test circuit (figure A.1) is approximately equivalent to two terminators creating the normal system bias.
2 Including the weak output.
3 The test limits shall be within the shaded area of figure A.2.

Table A.3 - Driver steady-state test limits and ~~conditions~~conditions for paced transfers

Test parameter	Test conditions (figure A.1)(note 1)	Minimum (mV) (note 2)	Maximum (mV)
$\lvert V_A \rvert$ Differential output voltage magnitude (asserted) (note)	V_1 = 1,056 V V_2 = 0,634 V	~~TBD~~370	800
	V_1 = 1,866 V V_2 = 1,444V	~~TBD~~370	800
$\lvert V_N \rvert$ Differential output voltage magnitude (negated) (note)	V_1 = 1,056 V V_2 = 0,634 V	~~TBD~~370	800
	V_1 = 1,866 V V_2 = 1,444V	~~TBD~~370	800
$\lvert V_A \rvert$ Differential output voltage magnitude (asserted)	~~All four above conditions~~	~~TBD~~ 0,90 x $\lvert V_N \rvert$ - 23	~~TBD~~ 1,11 x $\lvert V_N \rvert$ + 26

Notes:
1 The test circuit (figure A.1) is approximately equivalent to two terminators creating the normal system bias.
2 Including the weak output.
3 The test limits shall be within the shaded area of ~~figure A.2~~figure A.3.

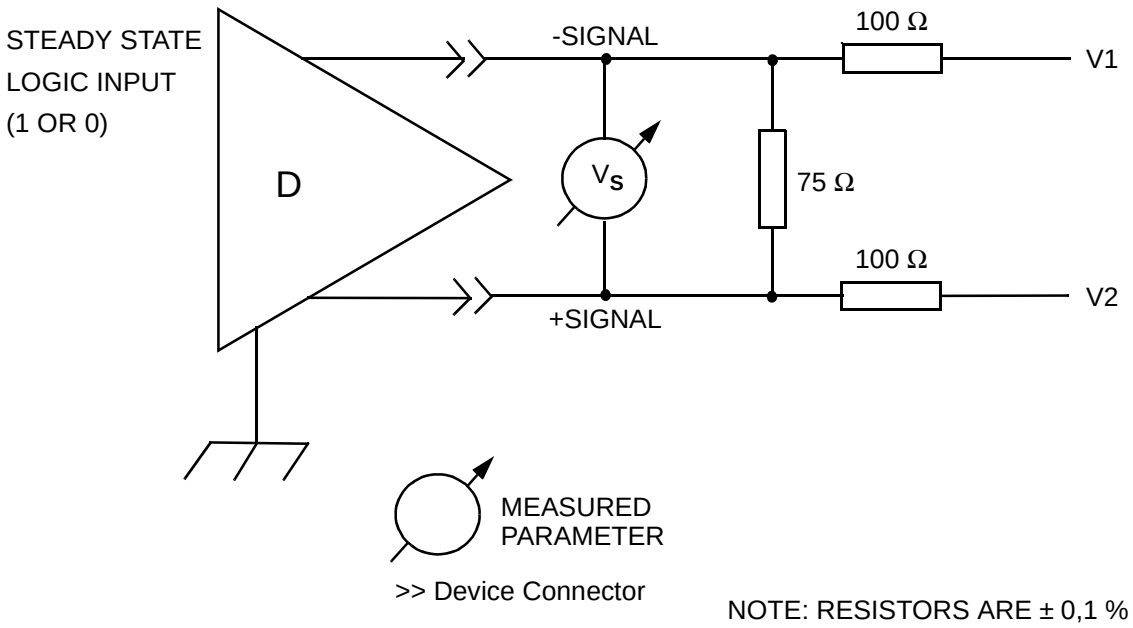


Figure A.1 - Differential steady-state output voltage test circuit

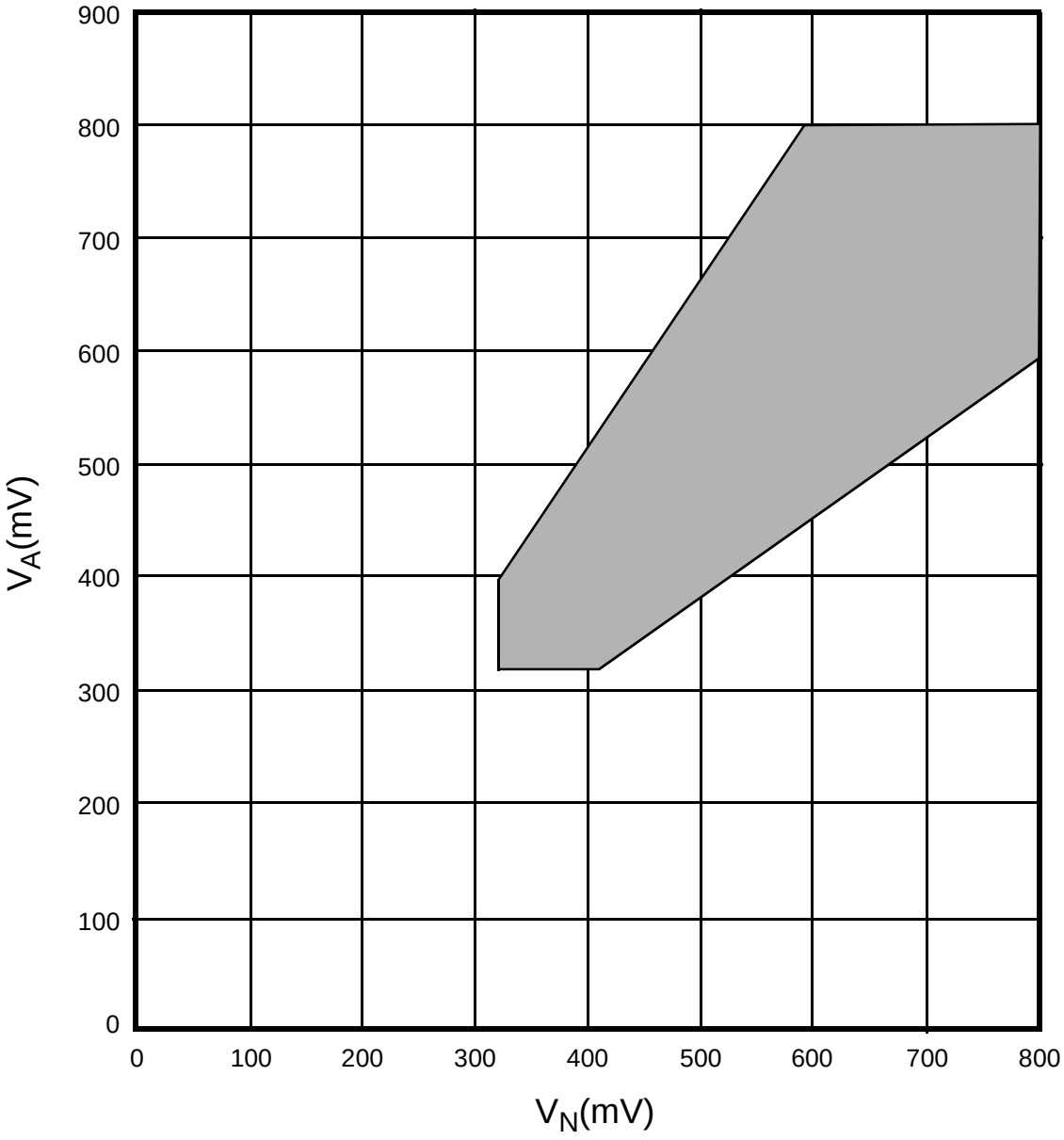


Figure A.2 - Domain for driver assertion and negation levels for synchronous transfers

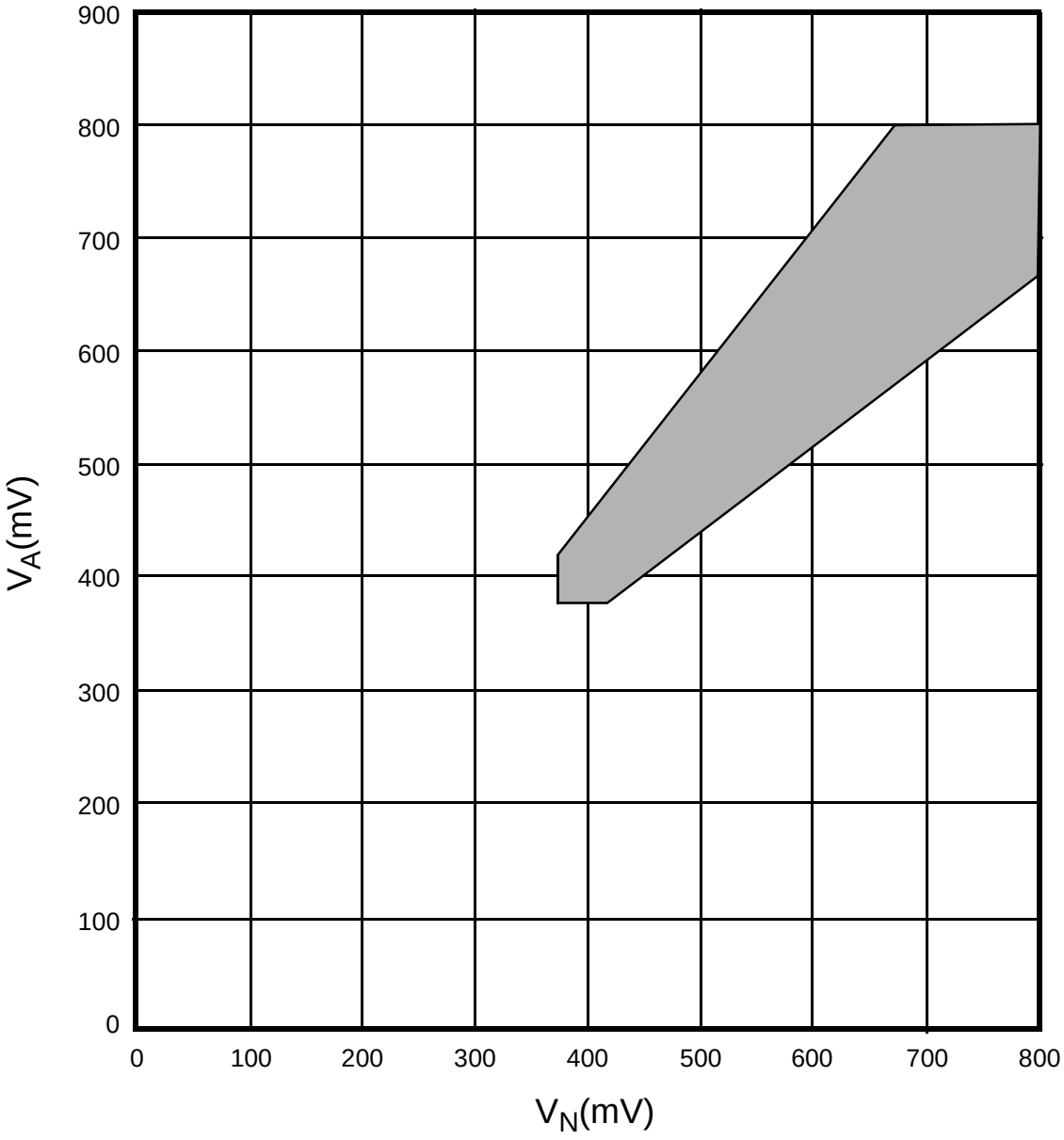


Figure A.3 - Domain for driver assertion and negation ~~levels~~levels for paced transfers

A.2.3 Offset (common-mode output) voltage, V_{CM}

The steady-state magnitude of the driver offset voltage (V_{CM}), measured with the test load of figure A.4 shall be greater than or equal to 0,845 V and less than or equal to 1,655 V for either binary state. The steady-state magnitude of the difference of V_{CM} for one logical state and for the opposite logical state, $\Delta V_{CM,}$ shall be 120 mV or less for all $V_{applied}$ in the range: $0{,}845 \leq V_{applied} \leq 1{,}655$. See figure A.5.

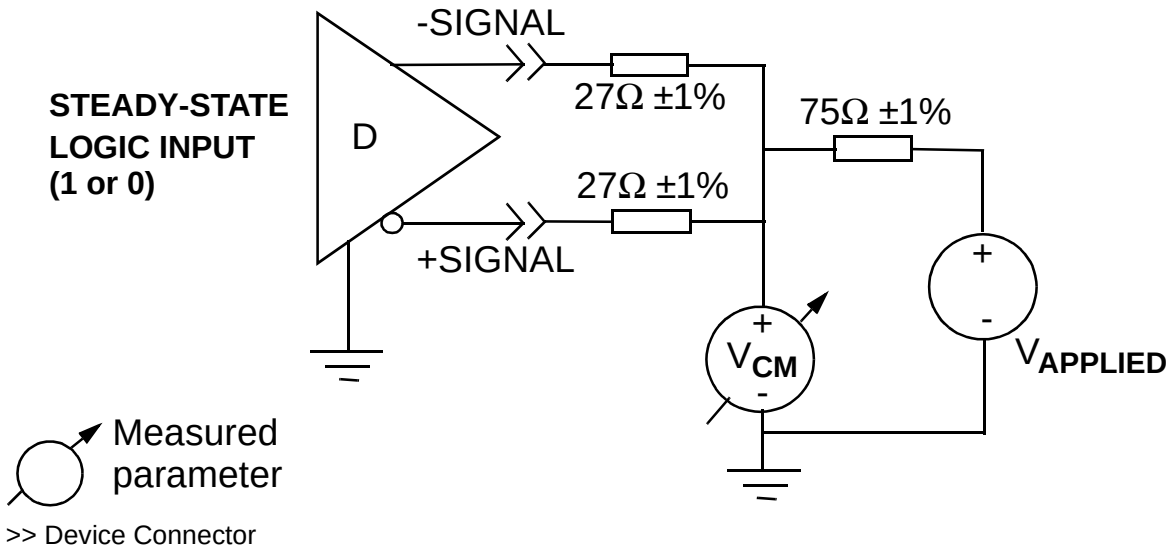


Figure A.4 - Driver offset steady-state voltage test circuit

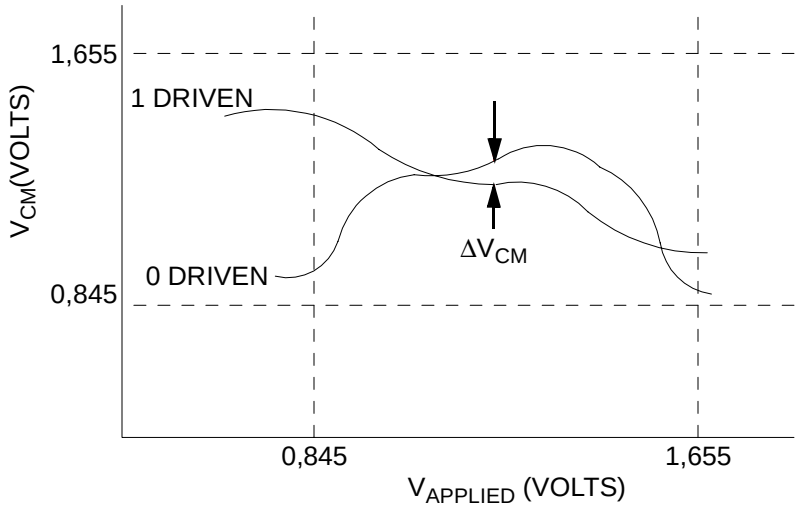


Figure A.5 - Common mode output voltage test

A.2.4 Short-circuit currents, I_{O-S} and I_{O+S}

Since an LVD bus allows multiple drivers, the possibility of contention requires a restriction on the power that may be sourced to the bus by a SCSI device. This is accomplished with a maximum allowable current from the driver.

With the driver output terminals short-circuited to a variable voltage source, the magnitudes of the currents (I_{O-S} and I_{O+S}) shall not exceed 24 mA for either logical state over the range $0 \leq V_{applied} \leq 2{,}5$ V. (see figure A.6).

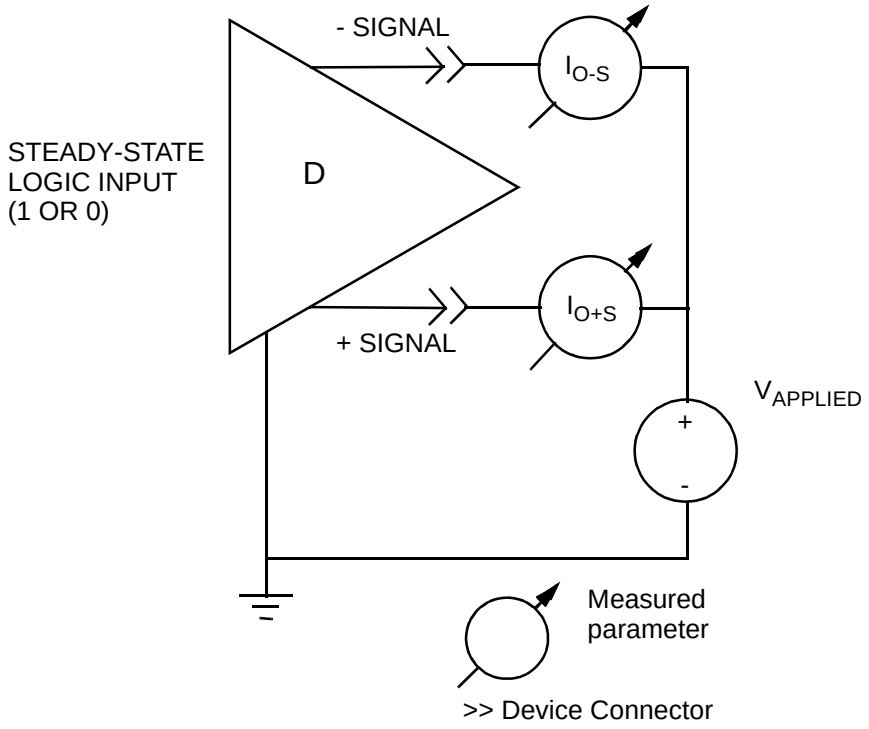


Figure A.6 - Driver short-circuit test circuit

A.2.5 Open-circuit output voltages, $V_{O-(OC)}$ and $V_{O+(OC)}$

To limit the maximum steady-state voltage at any device connector, the voltage between each output terminal of the driver circuit and its ground shall be between 0 V and 3,6 V when measured in accordance with figure A.7. This requirement shall be met in all logical or high impedance states (0 V ≤ $V_{O-(OC)}$ ≤ 3,6 V and 0 V ≤ $V_{O+(OC)}$ ≤ 3,6 V). The highest output voltage occurs with no output current.

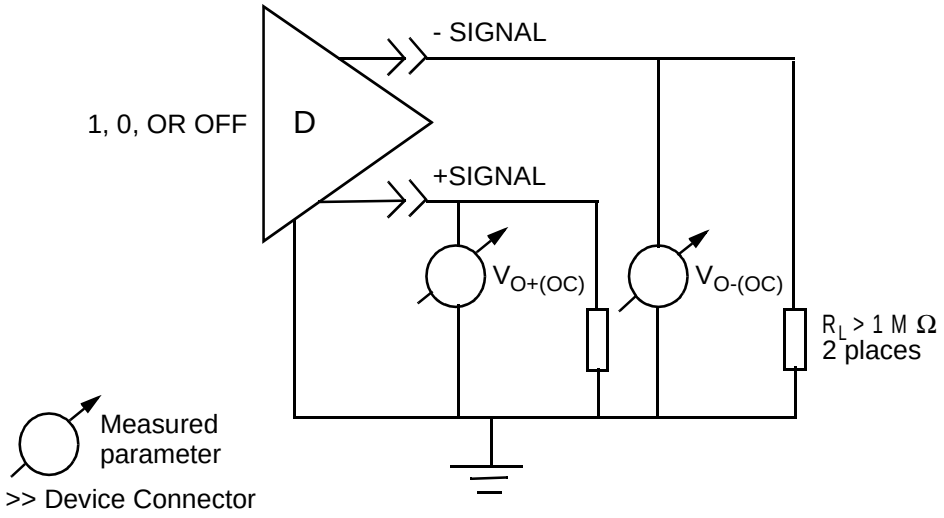


Figure A.7 - Open-circuit output voltage test circuit

A.2.6 Output signal waveform

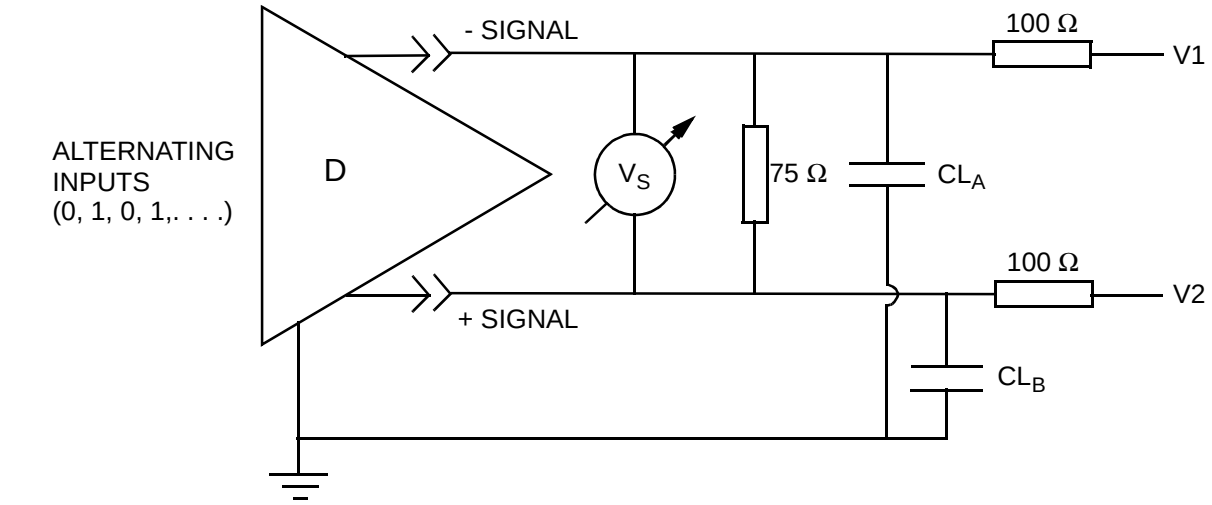
The differential output rise or fall time of a driver is specified since they influence the timing measurements and stub lengths of an LVD interface. Excessive over and under shoot of the output signal may cause electromagnetic emissions or false logic state changes.

During transitions of the driver output between alternating logical states (one - zero, zero - one, one - off, off - one, zero - off, off - zero), the differential voltage measured with the test circuit of figure A.8 and table A.4, shall be such that the voltage monotonically changes between 0,2 and 0,8 of the steady-state output, V_{SS}. V_{SS} is defined as the voltage difference between the two steady-state values of the driver output ($V_{SS} = |V_A| + |V_N|$) (See figure A.9 and table A.2). V_{SS} is expected to be different for different transitions.

The output signal rise or fall times (see t_r in figure A.9) between 0,2 and 0,8 of V_{SS} shall be greater than or equal to 1 ns. For paced transfers the output signal rise or fall times (see t_r in figure A.9) between 0,2 and 0,8 of V_{SS} shall be greater than or equal to 1 ns and less than or equal to 2,5 ns.

The rise and fall times specified above are requirements for a driver when using the LVD test circuit in figure A.8. They are not the observed rise or fall rates on an actual SCSI bus.

Measurement equipment used for rise and fall rate testing shall provide a bandwidth of 2 GHz minimum.



Notes:

a) Resistors are ±1% and surface-mount metal film type.
b) CL_A and CL_B are 5 pF ±0,2 pF and include the instrumentation capacitance.
c) The longest physical dimension between the device connector pins and any test circuit component shall be no greater than 0,1 meter.
d) V1 and V2 are applied voltages from a source having a source impedance of less than 5 Ω from 0 Hz to 40 MHz.

Figure A.8 - Differential output switching voltage test circuit

Table A.4 - Driver switching test circuit parameters

Test condition (see figure A.8)	V1	V2
Low common-mode voltage	1,311 V	0,889 V
High common-mode voltage	1,611 V	1,189 V

The signal voltage shall comply with the requirements shown in figure A.9.

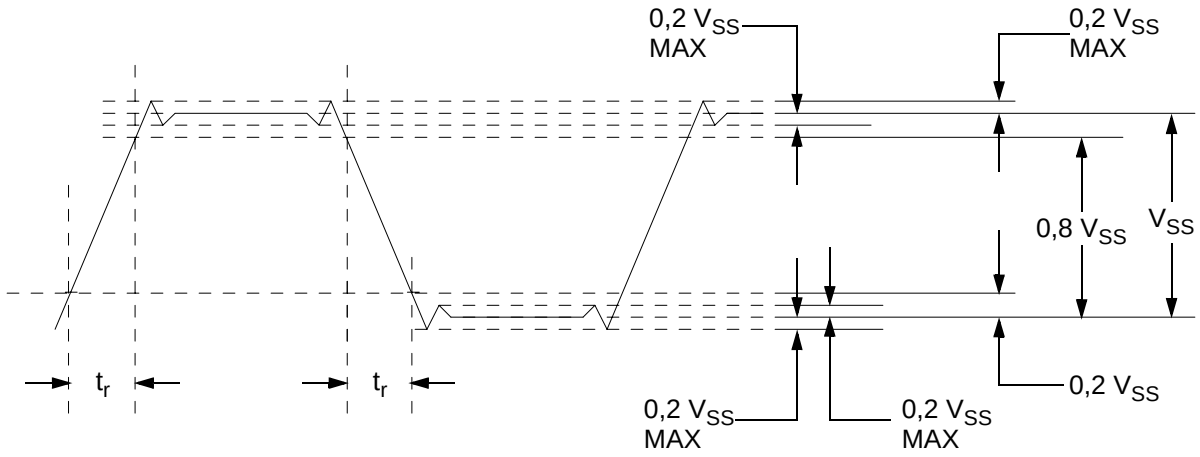
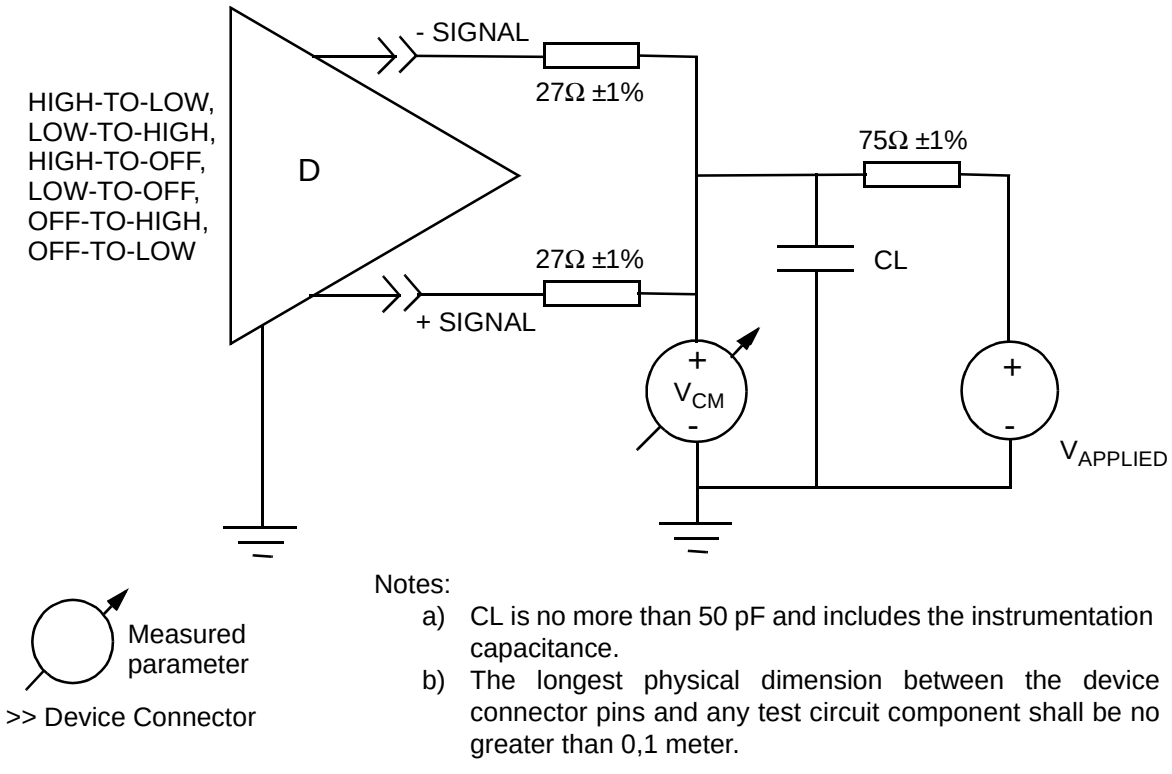


Figure A.9 - Driver output signal waveform

A.2.7 Dynamic output signal balance, $V_{CM(PP)}$

A mismatch in the magnitude of rate at which the voltage changes at the - signal and + signal connector pins, results in a common-mode AC signal. This may cause electromagnetic emissions from the media, excursions outside the receivers' common-mode input voltage range, and/or differential noise.

During transitions of the driver output between any state transition of high-to-low, low-to-high, high-to-off, off-to-high, low-to-off, or off-to-low, the voltage (V_{CM}) measured with the test circuit shown in figure A.10, shall not vary more than specified in table A.5 as $V_{applied}$ is varied over the range $0,845 \leq V_{applied} \leq 1,655$. Measurement equipment used for dynamic signal output balance testing shall provide a bandwidth of 400 MHz minimum. The requirements in this subclause apply only to the applicable state transitions.



Notes:

a) CL is no more than 50 pF and includes the instrumentation capacitance.

b) The longest physical dimension between the device connector pins and any test circuit component shall be no greater than 0,1 meter.

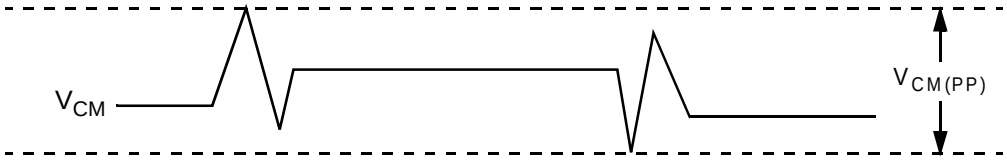


Figure A.10 - Driver offset switching voltage test circuit

Lower values of $V_{CM(PP)}$ have lower EMI risk.

Table A.5 - Dynamic output balance limits

Transition	$V_{CM(PP)}$ mV max
high-low	120
low-high	120
high-off	400
low-off	400
off-high	400
off-low	400

A.3 Receiver characteristics

A.3.1 Receiver characteristics overview

A receiver indicates the logical state of the LVD bus as defined by the differential voltage that exists at the device connector. A minimum steady state differential voltage defines the logic state. The receiver shall detect this difference over the allowable common-mode input voltage range as determined by the driver and terminator output offsets and ground difference voltages.

SCSI devices should incorporate a glitch filter function on REQ and ACK signals to reduce or eliminate the effect of glitch pulses.

If implemented, the glitch filter period shall not be so long as to mask out the subsequent valid transition edges of the incoming REQ and ACK signals.

A.3.2 Receiver steady state input voltage requirements

Table A.6 defines the voltages for the requirements in this subclause.

Within the common-mode input voltage range (V_{CM}), (figure 45) 0,845 V $\leq V_{CM} \leq$ 1,655 V an LVD receiver shall indicate the logical states shown in table A.6 with V_{IN} within the ranges shown in table A.6.

Table A.6 - Receiver steady state input voltage ranges

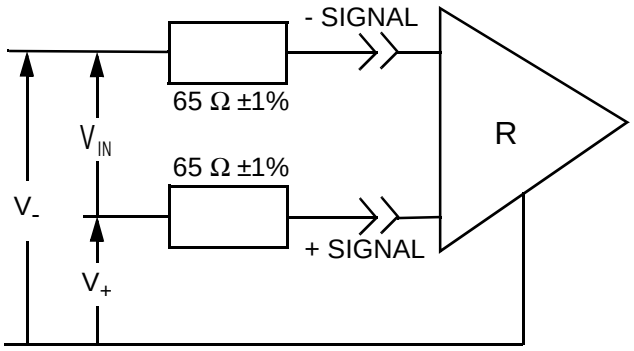
Differential Input voltage range steady state (V_{IN})	Receiver detects
-3,6 V $\leq V_{IN} \leq$ -0,030 V	1
0,030 V $\leq V_{IN} \leq$ 3,6V	0

A.3.3 Compliance test

Compliance to the requirements in A.3.2 shall be verified with the input voltages of table A.7 and the circuit of figure A.11.

Table A.7 - Receiver minimum and maximum input voltages.

Applied voltages (input voltage referenced to circuit ground) (see figure A.11)		Resulting differential input voltage	Resulting common-mode input voltage
V_-	V_+	V_{IN}	V_{CM}
0,860	0,830	0,030	0,845
0,830	0.860	-0.030	0,845
1,670	1,640	0,030	1,655
1,640	1,670	-0.030	1,655
3,310	0,000	3,310	1,655
0,000	3,310	-3,310	1,655
3,665	-0,355	4,020	1,655
-0,355	3,665	-4,020	1,655



>> Device Connector

Figure A.11 - Receiver input voltage threshold test circuit

A.3.4 Receiver setup and hold times

Figure 48 and figure 49 define the receiver setup and hold times.

NOTE 48 - Dynamic testing is required to verify these timings.

A.4 Transceiver characteristics

A.4.1 Transceiver output/input currents, I_{I-L} and I_{I+L}

The requirements in this subclause apply as a test method to ensure compliance with the LVD parameters table 15 and table 16. With the transceiver in an off condition (i.e., not transmitting) and the + and - signals connected to a variable voltage source, $V_{applied}$, the output leakage currents I_{I-L} and I_{I+L} shall not exceed the applicable LVD values in table 16 over the range 0,00 V ≤ $V_{applied}$ ≤ 3,01 V (see figure A.12). The maximum LVD applicable current from table 16 is I_{max}.

These measurements apply with the transceiver's power supply in both the powered on and powered off conditions.

$| I_{I-L} | < I_{max}$

$| I_{I+L} | < I_{max}$

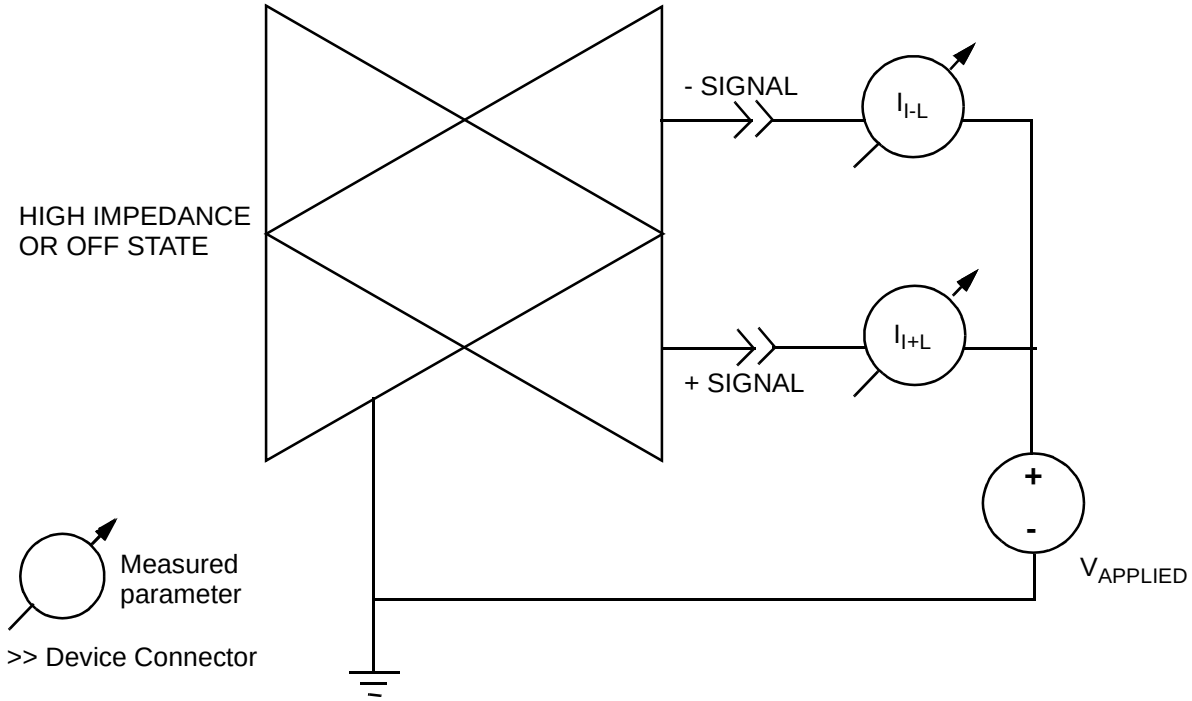


Figure A.12 - Transceiver off-state output current test circuit

A.4.2 Transceiver maximum input voltages

See table 15 and table 16.

Annex B

(normative)

SCSI bus fairness

B.1 Model

Implementation of the SCSI bus fairness is optional, however, if implemented, the SCSI bus fairness protocol shall conform to this annex.

A SCSI device determines fairness by monitoring prior arbitration attempts by other SCSI devices. It shall postpone arbitration for itself until all lower priority SCSI devices that previously lost arbitration either win a subsequent arbitration or discontinue their arbitration attempts (e.g., as in the case where the initiator aborted an outstanding command thus removing the need to re-arbitrate).

When a SCSI device does not need to arbitrate for the SCSI bus, it shall monitor the arbitration attempts of the other SCSI devices and refresh the fairness register with the SCSI IDs of any lower priority SCSI devices that lost arbitration.

Whenever a requirement for arbitration arises, a SCSI device shall first check to see if its fairness register is clear (see 10.5.1). If it is clear, then no lower priority SCSI device has attempted and lost the previous arbitration and therefore, this SCSI device may now participate in arbitration. If the fairness register is not clear, the SCSI device shall postpone arbitration until all lower priority SCSI IDs have been cleared from the fairness register. Lower SCSI IDs are cleared as those SCSI devices win arbitration. SCSI IDs shall also be cleared if a SCSI device discontinues arbitration (e.g., as a result of an ABORT TASK message, ABORT TASK SET message, CLEAR TASK SET message, TARGET RESET message, hard reset).

The fairness register may be refreshed, updated or cleared. The fairness register is refreshed by copying the SCSI IDs of any lower priority SCSI devices that lost arbitration into the fairness register. A refresh of the fairness register completely replaces the previous contents of the fairness register. The fairness register is updated by removing the SCSI IDs of any lower priority devices that win arbitration or discontinue arbitration. The fairness register is cleared by setting all of its bits to zero. SCSI IDs may only be added to the fairness register by a refresh but may be subtracted by a refresh, update or clear.

Since the fairness register is only refreshed when a SCSI device is not arbitrating for itself, the fairness register is effectively frozen by a SCSI device prior to a requirement for its own arbitration arising. Other lower priority SCSI devices that were not latched shall not be added to the fairness register until this SCSI device has successfully arbitrated.

Arbitration fairness in targets is controlled with the disconnect-reconnect mode page (see 18.1.2).

B.2 Determining fairness by monitoring prior bus activity

B.2.1 Fairness for normal arbitration method

For the normal arbitration method this standard requires that within a bus set delay of when BSY was first asserted, the SCSI ID of all arbitrating SCSI devices shall appear on the bus. A SCSI device shall sample the bus after this time, to detect SCSI devices that are arbitrating, which SCSI device won, and which SCSI devices lost. Since the lower priority SCSI IDs begin to disappear after an arbitration delay from the assertion of BSY, the data bus shall be sampled after a bus set delay but before an arbitration delay.

NOTE 49 - For ease of implementation, the sample window may begin when BSY is asserted to begin arbitration and extending until SEL is asserted. Sampling of the bus during this time should occur at a high enough rate to ensure multiple samples within this window.

B.2.2 Fairness for QAS

For QAS, after detection of a valid QAS REQUEST message, this standard requires that between 1 000 ns (QAS arbitration delay) and 1 490 ns (QAS arbitration delay+bus settle delay+2 deskew delays) after detection of the MSG, C/D, and I/O signals being false the SCSI ID for all arbitrating SCSI devices shall appear on the bus. The SCSI device shall sample the bus during this time, to detect which SCSI devices are attempting arbitration, which SCSI device won, and which SCSI devices lost. Since the lower priority SCSI IDs begin to disappear at 1 490 ns, a continuous sampling of the data bus between 1 000 ns and 1 490 ns is required.

B.3 Fairness algorithm

B.3.1 Fairness states overview

A SCSI device shall be in one of three fairness states. A SCSI device shall be in fairness wait state if it is waiting for a clear fairness register to participate in arbitration. A SCSI device shall be in the fairness participate state if it is participating in arbitration. A SCSI device shall be in the fairness idle state for all other conditions. A SCSI device shall enter the fairness idle state after any reset event. A SCSI device shall implement a lockout delay to prevent devices that stop arbitrating from causing deadlock.

B.3.2 Fairness idle state

A SCSI device shall be in fairness idle state if it does not need to arbitrate for the SCSI bus. The fairness register shall be refreshed during each arbitration while in the fairness idle state. The fairness register shall then contain the SCSI IDs of any lower priority arbitration losers. A SCSI device shall transition to the fairness wait state if it needs to arbitrate and the fairness register is not clear. A SCSI device shall transition to the fairness participate state if it needs to arbitrate and the fairness register is clear.

B.3.3 Fairness wait state

A SCSI device shall be in fairness wait state if it needs to arbitrate for the SCSI bus, but the fairness register is not clear. The fairness register shall be updated during each arbitration while in the fairness wait state. The updates subtract out arbitration winners and non-participants. The updates shall not add any new arbitration participants. A SCSI device shall transition to the fairness participate state once all of the bits in the fairness register have been cleared. A SCSI device shall transition to the fairness idle state if it no longer needs to arbitrate.

B.3.4 Fairness participate state

A SCSI device shall be in fairness participate state if it is participating in arbitration. A SCSI device can only be in fairness participate state if the fairness register is clear. If the SCSI device wins the arbitration it shall refresh the fairness register. If the SCSI device loses arbitration it shall keep the fairness register clear and remain in the fairness participate state. A SCSI device shall transition to the fairness idle state if it no longer needs to arbitrate.

B.3.5 Lockout delay

A SCSI device shall implement a lockout delay to prevent deadlock during normal arbitration. The lockout delay shall be at least 2 us and shall start once bus free has been detected. Each SCSI device shall clear its fairness register if no SCSI device starts arbitration before the lockout delay expires.

NOTE 50 - The lockout delay is required because all the SCSI devices may be waiting for other SCSI devices to start the arbitration process. Although rare, the following example is valid and may occur. Assume an initiator at SCSI ID 7 starts tasks in SCSI devices at SCSI IDs 0 and 1. After a while, SCSI devices 0 and 7 begin arbitration, 7 wins and SCSI device 0 is recorded in the fairness register of all SCSI devices. Assume that SCSI device 7, the arbitration winner, aborts its task on SCSI device 0. Both SCSI device 7 and 1 will now be waiting on SCSI device 0 to arbitrate but SCSI device 0 no longer needs to

arbitrate since its task has been aborted. When the lockout delay expires, SCSI devices 7 or 1 may begin arbitration.

NOTE 51 - The lockout delay is not required for QAS arbitration because the bus shall go bus free after a QAS arbitration delay. The fairness register shall be cleared because of mixed arbitration (B.3.6).

B.3.6 Mixed arbitration

The fairness register shall be cleared by all SCSI devices that detect a change in arbitration mode. If the previous arbitration was an QAS arbitration then the fairness register shall be cleared if bus free is detected. If the previous arbitration was a normal arbitration then all devices that detect the QAS arbitration shall clear the fairness register.

NOTE 52 - SCSI devices that do not support QAS may not detect the QAS arbitration and may not clear their fairness register.

B.4 Initiator fairness recommendations

Generally the initiator is the highest priority SCSI device on the bus. This guarantees the initiator wins arbitration and may overlap commands to multiple SCSI devices. To maintain this capability, the initiator should not implement fairness towards lower priority targets.

In the case of a multi-initiator system, the initiators should be the highest priority SCSI devices. However, in order to implement fairness between them, the higher priority initiator could implement fairness with the lower priority initiators only. This would require a second register in which a bit is enabled for each lower priority SCSI device to which a higher priority SCSI device shall be fair.

Annex C

(normative)

Nonshielded connector alternative 4

C.1 Nonshielded connector alternative 4 signal definitions

For the physical descriptions and usage guidelines for the nonshielded connector alternative 4 see Single Connector Attachments (SCA-2), EIA-700A0AE and SCA-2 Unshielded Connections, SFF-8451.

C.2 VOLTAGE and GROUND signals

Three voltage supplies and corresponding ground return signals are provided by the backplane connector to the SCSI device. Table C.1 provides the specifications for each of the voltage supplies.

NOTE 53 - The details of the actual SCSI device supply requirements need to be tailored for each SCSI device and enclosure combination to ensure appropriate supply voltage to the SCSI device.

Table C.1 - Voltage specification limits

Voltage	Number of pins	Number of grounds	Requirements on supply at the nonshielded connector alternative 4 connector	Current capability average/peak
12 V	3	3	12 V D.C.+5 % / -7 %	0/0 to 2,5 / 5 Amps
5 V	2	2 (note)	5 V D.C.±5 %	0/0 to 2 / 2,5 Amps
Opt 3.3 V	2	2 (note)	3,3 V D.C.±5 %	0/0 to 3 / 3 Amps
Note: The two logic level grounds are shared between +5 V D.C.and +3,3 V D.C.				

The peak current capability is measured during operation or initialization after voltages have stabilized at the operating level. Inrush currents are managed by the power supply during normal power on and by the CHARGE signals during hot plugging.

For each voltage, the current supplied to the SCSI device should be distributed as evenly as possible among the connecting pins.

The backplane power supplies are required to operate correctly and maintain regulation from zero current to the peak current. SCSI Device sequencing provisions may be required to avoid overloading power supplies during SCSI device spin-up sequencing. Voltage dips to -10 % are allowed on the 12 V D.C.supply during spin up.

For each voltage, an appropriate number of current return GROUND signal pins have been assigned.

a) The GROUND signal pins for all voltages shall be tied together in the SCSI device.
b) The GROUND signals in the backplane may be tied together or connected separately to the power supplies as required by the particular subsystem.

c) The logic level grounds, GROUND (5V/3,3V) are shared between the currents provided by the 5 V D.C. and 3,3 V D.C. signals. The sum of the 5 V D.C. and 3,3 V D.C. currents shall not exceed 3 Amps.

C.3 CHARGE signals

Three charge signals, one for each of the power supply voltages, provide controlled precharging of the SCSI device's internal circuits to avoid excessive surge currents during hot plugging.

The precharge pin mates early to allow the precharge to take place before the voltage pins make contact. If required, the precharge control circuits are located on the backplane side of the connector. The backplane should assume that the VOLTS signals for each voltage are shorted together with the corresponding CHARGE signal on the SCSI device. Systems without a hot-plug capability or with an alternative hot plugging mechanism are not required to implement the precharge control circuit and are not required to use long and short pins on the backplane connector.

After the SCSI device capacitance is charged, but before the MATED signal indicates that the power signals are seated, the SCSI device shall not use more than 1 Amp on the precharge voltage pin. This is required to protect the precharge pin from over-current damage and to provide additional flexibility in the design of the precharge circuit. The voltage provided by the precharge circuitry is as specified by table C.2. Any circuitry on the SCSI device that uses the CHARGE voltage for executing initialization operations shall operate within the current and voltage constraints specified for the CHARGE signals.

Table C.2 - Charge supply to SCSI device

CHARGE signal	Requirements on supply at the nonshielded connector alternative 4 connector for backplane after CHARGE complete	Maximum surge at SCSI device	Maximum continuous required by SCSI device
12V	12 V D.C.+5 % / -12 %	6 Amps	1 Amp
5V	5 V D.C.+5 % / -17 %	6 Amps	1 Amp
OPT 3,3V	3,3 V D.C.+5 % / -24 %	6 Amps	1 Amp
Note: It may not be possible to meet these parameters with passive components.			

After precharge is complete and the SCSI device is mated, there is no guarantee that the precharge signal provides any current to the SCSI device and the SCSI device should not depend on such current for operation.

The system designer should assume that the VOLTS signal(s) and the corresponding CHARGE signal are shorted together on the SCSI device.

C.4 SPINDLE SYNC

The spindle synch is assigned a single pin, SPINDLE SYNC. The synchronization protocol and the electronic requirements for the SPINDLE SYNC signal are defined in the SCSI device’s specification.

Spindle synchronization is managed by the SCSI command set. The signal current requirements shall not exceed 100 milliamperes and the signal voltage shall not be higher than 5,25 V D.C.or lower than -0,25V. The minimum driver capability required by the SPINDLE SYNC signal shall be sufficient to drive the receivers on 30 identical SCSI devices.

The SPINDLE SYNC signal when driving should be capable of driving a minimum of 30 identical SCSI devices.

NOTE 54 - The SPINDLE SYNC signal is a source for noise and may be affected by noise. The design of the SPINDLE SYNC signal interconnections should take this into account by properly laying out the SPINDLE SYNC signals on the backplane or motherboard. Proper layout considers routing relative to other signals, the proper line impedance, and termination if necessary. The selection of the electronic transceiver should also take into account the possibility of noise. The signal levels, signal rise time, receiver thresholds, and receiver hysteresis should be considered as part of that selection.

C.5 ACTIVE LED OUT

The ACTIVE LED OUT signal is driven by the SCSI device when a SCSI operation is being performed. The ACTIVE LED OUT signal shall be implemented and is used to indicate that the SCSI device is operating on a command. Other optional indications may be provided by flashing the LED. The host system is not required to generate any visual output when the ACTIVE LED OUT signal is raised, but if such a visual output is provided, it shall be white or green to indicate that normal activity is being performed.

The ACTIVE LED OUT signal is designed to pull down the cathode of an LED. The anode shall be attached to the proper +5 V D.C.supply through an appropriate current limiting resistor. The LED and the current limiting resistor are external to the SCSI device.

See table C.3 for the output characteristics of the ACTIVE LED OUT signal

Table C.3 - Output characteristics of drive ACTIVE LED OUT signal

State	Current drive available	Output voltage
Drive LED off	$0 < I_{OH} <$ 100 µA	Not defined by this standard
Drive LED on	$I_{OL} >$ 30 mA	$0 < V_{OL} <$ 0,8V

C.6 Motor start controls

The method of starting the SCSI device's motor is established by the signals RMT_START and DLYD_START, as described in table C.4. The state of these signals may either be wired into the backplane socket or driven by logic on the backplane.

Each SCSI device location should have these signals supplied independently to ensure proper operation. If the signals were bussed, a SCSI device with a power failure may clamp the signals in a condition that caused operational SCSI devices to behave incorrectly.

a) If the GROUND state is implemented for RMT_START, bussing between SCSI devices is permissible.
b) If the GROUND state is implemented for DLYD_START, bussing between SCSI devices is permissible.
c) If the OPEN state is implemented for RMT_START, this signal shall not be bussed between SCSI devices.

d) If the OPEN state is implemented for DLYD_START, this signal shall not be bussed between SCSI devices.

Table C.4 - Definition of motor start controls

Case	DLYD_START	RMT_START	Motor Spin Function
1	open	open	Motor spins up at D.C. power on.
2	open	ground	Motor spins up only when START UNIT command is received.
3	ground	open	Motor spins up at D.C. power on after a delay in seconds 12 times (note) the value of the numeric SEL_ID for the SCSI device.
4	ground	ground	Reserved. SCSI devices not implementing this option shall execute power control according to the rules of case 2 (see Annex D).
Note: This value may be reduced by SCSI device suppliers to reflect the worst case time duration of peak current drains at the 12 V D.C.or 5 V D.C.source (or both) during motor spin up. In no case should the delay exceed 12 seconds.			

The OPEN and GROUND states are established as described in table C.5.

Table C.5 - Electronic requirements for input controls

State	Current	Voltage
open	0 µA < I_{IH} < 100 µA	2,4 V D.C.< V_{IH} < V_{CC} + 0,5V
ground	-3 mA < I_{IL} < 0 mA	-0,5 V D.C.< V_{IL} < 0,4V
Note: The SCSI device provides the voltage source for the open signal state.		

C.7 SCSI ID selection

The SCSI device address of the attached SCSI device shall be determined by the state of the signals SCSI ID(0-3). Table C.6 indicates the relationship between the level of the SCSI ID signals and the selected SCSI device address.

Table C.6 - SCSI device ID selection signals

Address	ID (0)	ID (1)	ID (2)	ID (3)
0	open	open	open	open
1	ground	open	open	open
2	open	ground	open	open
3	ground	ground	open	open
4	open	open	ground	open
5	ground	open	ground	open
6	open	ground	ground	open
7	ground	ground	ground	open
8 (note)	open	open	open	ground
9 (note)	ground	open	open	ground
10 (note)	open	ground	open	ground
11 (note)	ground	ground	open	ground
12 (note)	open	open	ground	ground
13 (note)	ground	open	ground	ground
14 (note)	open	ground	ground	ground
15 (note)	ground	ground	ground	ground

Note: Addresses in the range of 8 to 15 are only supported by SCSI devices implementing wide data transfers.

The OPEN and GROUND states are established as specified in table C.5.

C.8 MATED signals

C.8.1 MATED signals overview

If MATED 1 and MATED 2 signals are not mated then one or more short pins are not mated.

If MATED 1 and MATED 2 signals are mated then the mated condition of the short pins is indeterminate.

The MATED 1 and MATED 2 signals may indicate to the SCSI device that the SCSI device is seated in an nonshielded connector alternative 4 connector and it may begin power on processing. The circuit

described in figure C.1 or a similar circuit is used to implement the MATED function. The signal requirements are indicated below, but may be met by the example circuit or by similar circuits.

C.8.2 MATED 2/drive side

The signal is attached to signal ground on the SCSI device side.

C.8.3 MATED 2/backplane side

The signal is attached either directly or through optional logic in such a manner that the MATED 1 signal is held to a ground level when the MATED 2 connection is completed. The SCSI device shall sink no more than 100 mA to ground through the MATED 2 pin if optional logic is used.

C.8.4 MATED 1/drive side

The MATED 1 signal shall be sensed by the SCSI device. When the MATED 1 connection is determined to be at a ground level, the SCSI device may detect that the SCSI device has been partially mated. Assuming the mating process continues uninterrupted until competition, including sensing of the SCSI ID Selection signals and the motor start controls, then normal power on procedures may begin 250 msec after the MATED 1 signal is observed to transition to the ground level. When the MATED 1 connection is determined to be at the open level, the SCSI device is not mated. The MATED 1 signal is tied up to a TTL positive level when the SCSI device is not installed.

If the SCSI device is mated and operating, it may optionally detect the open level of MATED 1 as an indication that the SCSI device is partially unmated and may be about to be removed.

If the SCSI device supports detection of the open level of MATED 1 to prepare itself for power removal or for physical removal from the enclosure, the detection shall occur within 1 second from the time that the Mated 1 open level occurs at the SCSI device.

The following SCSI device behaviors are defined when a SCSI device detects the open level of MATED 1:

a) The SCSI device may optionally perform a spin-down operation.
b) The SCSI device may optionally transfer any cached information to the media.

C.8.5 MATED 1/backplane side

The signal shall be held to a ground level when the MATED 2 connection is completed. The MATED 1 signal shall be held to the open level when the MATED 2 connection is not completed. The ground and open levels are defined by table C.5.

The enclosure may optionally control the MATED 1 signal to indicate that the SCSI device is about to be removed.

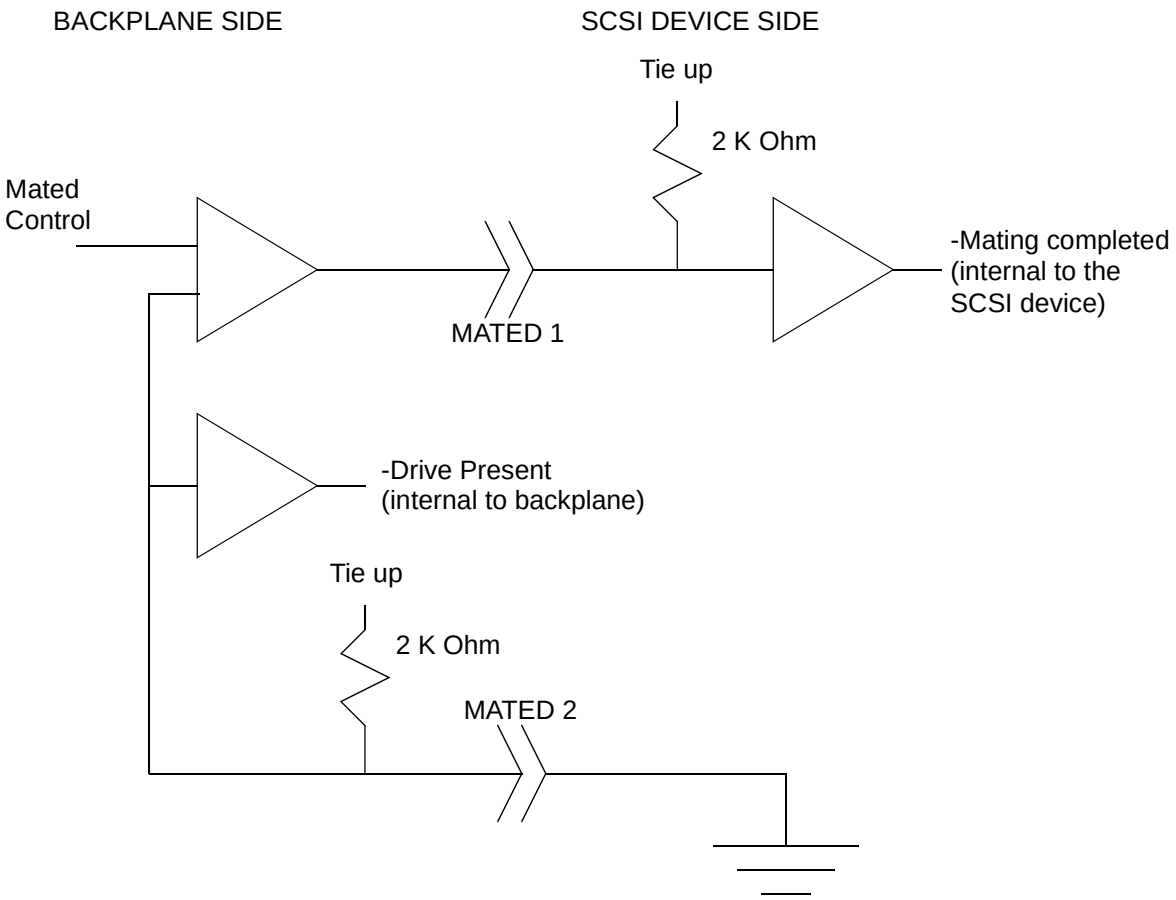


Figure C.1 - Sample circuit for mated indications

Annex D

(normative)

Removal and insertion of SCSI devices

D.1 Removal and insertion of SCSI devices overview

This annex defines the physical requirements for removal and insertion of SCSI devices on the SCSI bus. The issues related to the logical configuration of the SCSI bus and characteristics of the SCSI devices when a replacement occurs are beyond the scope of this standard. It should also be noted that the cases listed are distinguished for compatibility reasons and in most cases describe a system environment independent of this standard.

Four cases are addressed. The cases are differentiated by the state of the SCSI bus when the removal or insertion occurs.

D.2 Case 1 - Power off during removal or insertion

a) All SCSI devices are powered off during physical reconfiguration.

D.3 Case 2 - RST signal asserted continuously during removal or insertion

a) RST signal shall be asserted continuously by the initiator during removal or insertion.
b) The system shall be designed such that the SCSI device being inserted shall make its power ground and logic ground connections at least 1 ms prior to the connection of any device connector contact to the SCSI bus. The ground connections shall be maintained during and after the connection of the SCSI device to the SCSI bus;
c) The system shall be designed such that the SCSI device being removed shall maintain its power ground and logic ground prior to, during, and for at least 1 ms after the disconnection of any device connector contact from the SCSI bus.

NOTE 55 - The translation of the 1 ms time to mechanical provisions is vendor specific.

D.4 Case 3 - Current I/O processes not allowed during insertion or removal

a) All I/O processes for all SCSI devices shall be quiesced;
b) The system shall be designed such that the SCSI device being inserted shall make its power ground and logic ground connections at least 1 ms prior to the connection of any device connector contact to the SCSI bus. The ground connections shall be maintained during and after the connection of the SCSI device to the SCSI bus;
c) The system shall be designed such that the SCSI device being removed shall maintain its power ground and logic ground prior to, during, and for at least 1 ms after the disconnection of any device connector contact from the SCSI bus;
d) The SCSI device being removed or inserted shall employ transceivers that conform to the applicable requirements in 7.2.2 and 7.3.5.3 for glitch-free powering on and off. The SCSI device shall maintain the high-impedance state at the device connector contacts during a power cycle until the transceivers are enabled. Power cycling includes on-board TERMPWR cycling caused by plugging, and SCSI device power cycling caused by plugging and switching;

NOTE 56 - Any on board switchable terminators as well as SCSI device transceivers may affect the impedance state at the device connector contacts.

e) The SCSI device power may be simultaneously switched with the SCSI bus contacts if the power distribution system is able to maintain adequate power stability to other SCSI devices during the

transition and the grounding requirements in items (b) and (c) above are met;
f) The SCSI bus termination shall be external to the SCSI device being inserted or removed.
g) Resumption of I/O processes is vendor specific but shall not occur sooner than 200 milliseconds after the completion of the insertion or removal event.
h) Bypassing capacitors connecting to the TERMPWR line on the SCSI device being inserted or removed shall not exceed 10 µF. For single-ended applications, SCSI bus terminations shall use voltage regulation.

NOTE 57 - In a multimode environment any insertion or removal that changes the SCSI bus mode causes a transceiver mode change reset event (see 12.4.2).

D.5 Case 4 - Current I/O process allowed during insertion or removal

a) All I/O processes for the SCSI device being inserted or removed shall be quiesced prior to removal.
b) A SCSI device being inserted shall make its power ground and logic ground connection at least 1 ms prior to the connection of any device connector contact to the SCSI bus. The ground connections shall be maintained during and after the connection of the SCSI device to the SCSI bus;
c) A SCSI device being removed shall maintain its power ground and logic ground prior to, during, and for at least 1 ms after the disconnection of any device connector contact from the SCSI bus;
d) The SCSI device being removed or inserted shall employ transceivers that conform to the applicable requirements in 7.2.2 and 7.3.5.3 for glitch-free powering on and off. The SCSI device shall maintain the high-impedance state at the device connector contacts during a power cycle until the transceivers are enabled. Power cycling includes on board TERMPWR cycling caused by plugging, and SCSI device power cycling caused by plugging and switching;

NOTE 58 - Any on-board switchable terminators as well as SCSI device transceivers may affect the impedance state at the device connector contacts.

e) The SCSI device power may be simultaneously switched with the SCSI bus contacts if the power distribution system is able to maintain adequate power stability to other SCSI devices during the transition and the grounding requirements in items (b) and (c) above are met;
f) The SCSI bus termination shall be external to the SCSI device being inserted or removed;
g) Initiation or resumption of I/O processes for a newly inserted or removed SCSI device is vendor specific but shall not occur sooner than 200 milliseconds after the completion of the insertion or removal event.
h) Bypassing capacitors connecting to the TERMPWR line on the SCSI device being inserted or removed shall not exceed 10 µF. For single-ended applications, SCSI bus terminations shall use voltage regulation.

NOTE 59 - In a multimode environment any insertion or removal that changes the bus mode causes a transceiver mode change reset event (see 12.4.2).

NOTE 60 - LVD SCSI devices may require more stringent system design to tolerate transients that occur during case 4 insertion or removal.

Annex E

(normative)

SCSI cable media performance testing

E.1 Overview

This annex defines the electrical measurement methodology requirements to be used for shielded and unshielded cable media. These methods are required to minimize the error between measurements executed in different laboratories.

The methodologies are specified to extract parameters for each of the following performance requirements:

a) Transmission line impedance (Z_O),
b) capacitance,
c) dielectric constant variation,
d) propagation time,
e) propagation time skew,
f) attenuation, and
g) near-end crosstalk.

Some parameters are specified for both single ended and differential applications.

Table E.1 summarizes the measurement methodology requirements.

Table E.1 - Cable media test summary

Parameter	Subclause	Measurement domain	Test conditions	Sample configuration	Active equipment
Single ended local Z_O	E.2.1	Time	Rise time at 3 ns	(A) tie one wire of each pair & shield together, 3 m long	TDR (e.g., Tektronix 11801 or equivalent)
Differential local Z_O	E.2.2	Time	Rise times at 1 ns and 3 ns	(B) all wires + shield floating, 3 m long	TDR (e.g., Tektronix 11801 or equivalent)
Differential extended distance (balanced) Z_O	E.2.3	Frequency	sweep between 30 MHz and 600 MHz	(C) same as (B) with minimum 30 m length	HP 8753E network analyzer or equivalent
Single ended capacitance	E.3.1	Frequency	100 KHz and 1 MHz	(D) Sample length 3 m	LCR meter
Dielectric constant variation with frequency	N/A	Frequency	Sweep between 300 KHz and 600 MHz	(E) Sample of one pair approximately 2,54 cm in length	Network analyzer
Differential capacitance	E.3.2	Frequency	1 MHz and 120 MHz	(F) Sample length 3 m	LCR meter
T_p (differential propagation) time per meter (note 1)	E.4	Frequency	An S_{12} measurement (note 4) swept from 10 MHz to 650 MHz (normative measurement with no pass-fail levels)	(G) Sample 30 m	HP 8753E network analyzer or equivalent

Notes:
1 - T_p is 1 / V_p
2 - Propagation time is measured at the amplitude mid-point of the Signal Transition Duration (STD)
3 - low frequency shelf: That range where a 3 times change in frequency produces less than 0,5 dB in attenuation difference
4 - Is a scatting parameter that relates the incident and transmitted voltage waves in a two-port circuit.

Table E.1 - Cable media test summary

Parameter	Subclause	Measurement domain	Test conditions	Sample configuration	Active equipment
T_p (differential propagation) time per meter (note 1)	E.5	Time	Launched rise time between 1 ns and 3 ns (note 2)	(G) Sample 6 m	Signal pulse, (no signal generator) TDR 11801 or equivalent
Differential propagation time skew	E.6		Difference between the minimum and maximum T_p of all pairs		N/A
Differential attenuation (balanced)	E.7	Frequency	low frequency shelf (note 3) to 1 GHz	(I) Sample leave all other lines open - long enough to produce at least 1 dB at the low frequency shelf (note 3) (typically > 30 m)	HP 8753E network analyzer or equivalent
Crosstalk NEXT Differential (Balanced)	E.8	Time	Signal pulse, maximum legal size, minimum signal transition duration time	(I) Sample For frequency option use sample set up (G)	Tektronix 11801 with TDR or equivalent

Notes:
1 - T_p is 1 / V_p
2 - Propagation time is measured at the amplitude mid-point of the Signal Transition Duration (STD)
3 - low frequency shelf: That range where a 3 times change in frequency produces less than 0,5 dB in attenuation difference
4 - Is a scatting parameter that relates the incident and transmitted voltage waves in a two-port circuit.

Methods provided use either frequency or time domain measurements as noted in table E.1. Attempts are made to cover as much of the application as practical. Specifically, both local and extended distance parameters and the range of applicable frequencies are required.

The physical construction details may affect the measured performance parameters. For example, using similar materials a planar discrete construction intrinsically has higher near end crosstalk than twisted pair construction. However, properly constructed planar cable media may perform well for SCSI. This annex does not specify any construction requirements but rather relies on the measured performance results as the criteria for compliance.

Several test parameters use a per unit length measurement in order to allow concatenation of different lengths of cable without exceeding the performance limits at maximum length.

Cable media provides only part of the electrical path in system applications. The requirements in this annex only apply to cable media performance in nominally uniform media.

Media constructions designed to be non-uniform for purposes of enabling connectorization are not covered in this annex. An example of such media is unshielded round media that has areas where the conductors are constrained to be straight (in a line) for short distances so that an insulation displacement type of connector may be attached. These types of media are considered to be part of a cable assembly (where connectors are attached) and the performance may not be accurately assessed without also considering the connectors.

In system applications effects not specified in this annex, but nevertheless related to cable media, may affect the bus performance. For example, the use of cable media with different impedance values in the same bus should be avoided to minimize discontinuities and signal reflections.

Other effects that may affect the performance of a complete SCSI bus segment, for example:

a) spacing of media conductors from other physical structures for non-shielded constructions (e.g. wires in other media, metallic walls, non-metallic surfaces);
b) non uniform device loading across all the SCSI signals;
c) non uniform stub properties;
d) the population and spacing of devices;
e) connectors and media disturbances required to attach connectors; and
f) data phase speed.

E.2 Impedance

E.2.1 Local impedance for SE transmission

E.2.1.1 Local impedance for SE transmission overview

The impedance measurement produces a plot of transmission line impedance as recorded by a time domain reflectometer instrument. There is a direct mapping of the measurements to the physical position within the cable under test. The test shall be performed at the specified signal transition duration time for the signals being used in the end-user application.

The sample length is long enough to ensure no interference from the far end.

E.2.1.2 Sample preparation

This test requires type A samples (see table E.1) prepared in the following way:

1) Cut sample length to $3 \pm 0{,}025$ m.
2) Remove 5,0 cm of outer jacket from one end.
3) Comb out braid wire strands to form a pigtail.
4) Trim filler and tape materials to the base of braid wire.
5) Strip 0,5 cm insulation from all conductors.
6) Connect one wire from each pair to the shield (for planar type cable, tie every other conductor to ground).

E.2.1.3 Test fixture and measurement equipment

Figure E.1 shows the test configuration for the local single ended impedance test.

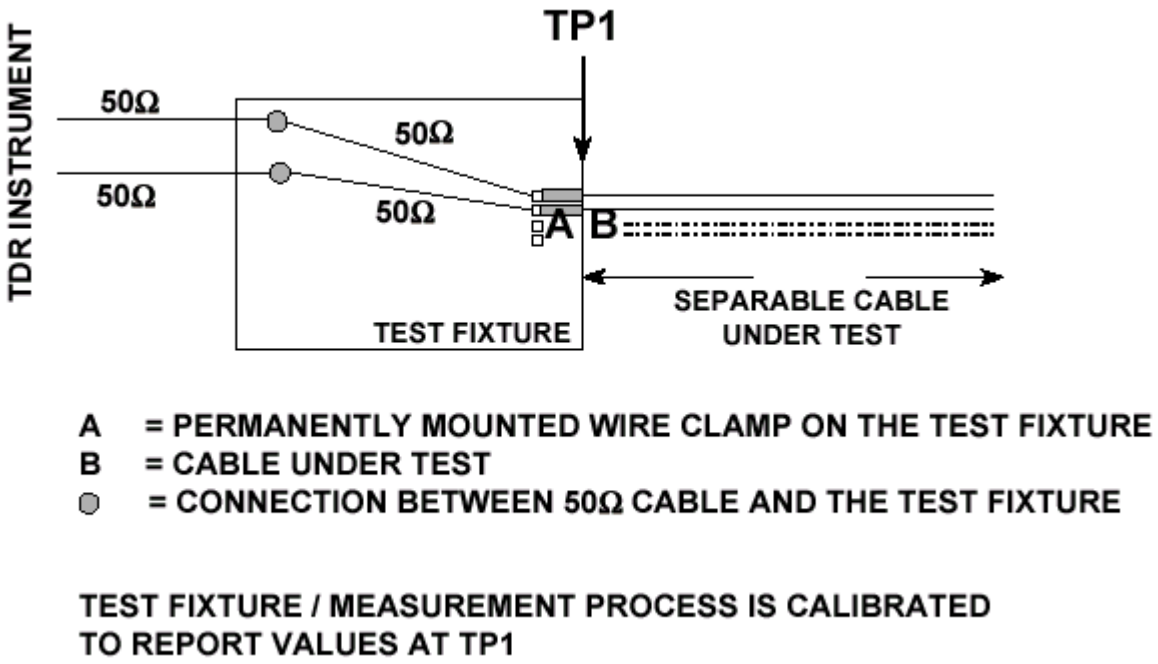


Figure E.1 - Test configuration for SE impedance

The test fixture may be constructed of semi-rigid coax, microstrip PCB, or stripline PCB. To be able to see the entire signal transition it is necessary that the length from the TDR to the DUT be long enough to contain the complete transition. For the 3 ns STD this minimum length is approximately 1,25 m for cables with solid polyethylene dielectric. For other dielectrics this length should be adjusted appropriately.

E.2.1.4 Calibration and verification procedure

E.2.1.4.1 Instrument verification

It is not necessary to perform a separate instrument verification for this test. The calibration in the following section includes the instrument.

E.2.1.4.2 Measurement system (with test fixture) calibration

Connect the 50 Ω cable to the test fixture. In place of “B” in figure E.1, connect a 100 Ω ± 0,1 % (preferred) low inductance chip resistor (i.e., IMS style TPI-1206 or equivalent). Use an unfiltered trace and the TDR cursors to measure the resistance value, R100, approximately 4 ns (displayed) after the resistor discontinuity. See figure E.2.

In a similar manner, in place of “B” in figure E.1, connect a 75 Ω ± 0,1 % (preferred) low inductance chip resistor. Use an unfiltered trace and use the TDR cursors to measure the resistance value, R75, approximately 4 ns (displayed) after the resistor discontinuity.

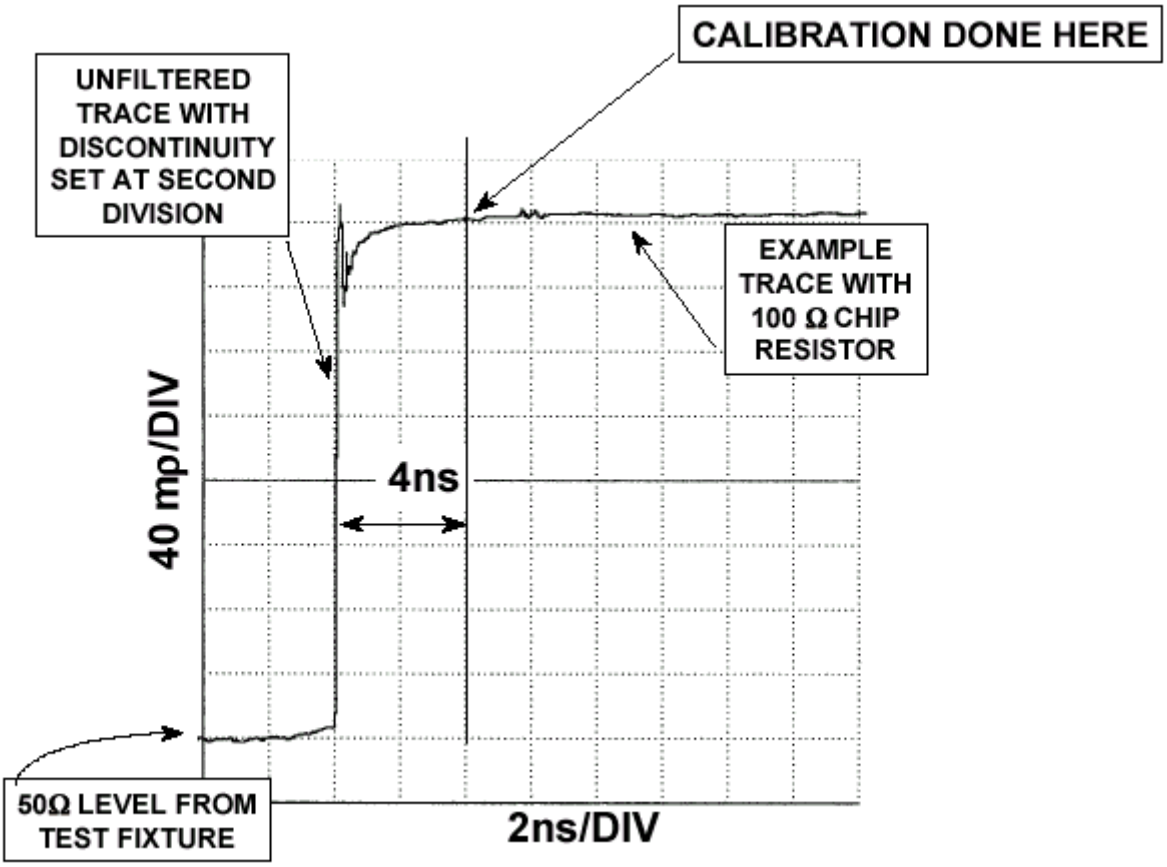


Figure E.2 - SE impedance calibration

For R100 and R75 the equation for determining the corrected (i.e., actual) impedance is:

$$Z_{corrected} = \frac{25 \times (((4 \times X_1) - (3 \times X_2)) - Z_{mesasured})}{X_1 - X_2}$$

Where:

a) X_1 is the measured value using the 75 Ω resistor
b) X_2 is the measured value using the 100 Ω resistor

E.2.1.4.3 Single ended signal transition duration (STD) calibration

This step ensures that the proper signal transition (STD) time is being presented to the DUT. Place a short on the test fixture where the cable would be attached in place of “B” in figure E.1. Use the filter function on the TDR to set the measured STD (20 % - 80 %) to the required value according to the detailed procedure described below. It may be desirable to use a separate test fixture that is nominally identical to the actual cable test fixture for this step.

Assuming a falling edge, set up the display on the TDR as shown in figure E.3. This display has the following properties:

a) The time scale on the display is 2,0 ns / div for the 3,0 ns STD.
b) Set the horizontal position such that the midpoint of the displayed curve is near the center of the display and the 100 % and 0 % baselines are clearly visible as shown in figure E.3.

The STD is the time between the 20 % and 80 % values of the displayed signal amplitude (most instruments do this calculation automatically). When the instrument does not automatically measure STD, perform the following steps:

1) Measure the voltage at 100 % (V_{100}).
2) Measure the voltage at 0 % (V_0).
3) The voltage at 20 % is $V_{20} = V_0 + 0{,}2(V_{100} - V_0)$.
4) The voltage at 80 % is $V_{80} = V_0 + 0{,}8(V_{100} - V_0)$.
5) Set cursors at V_{20} and V_{80} and measure the time difference.

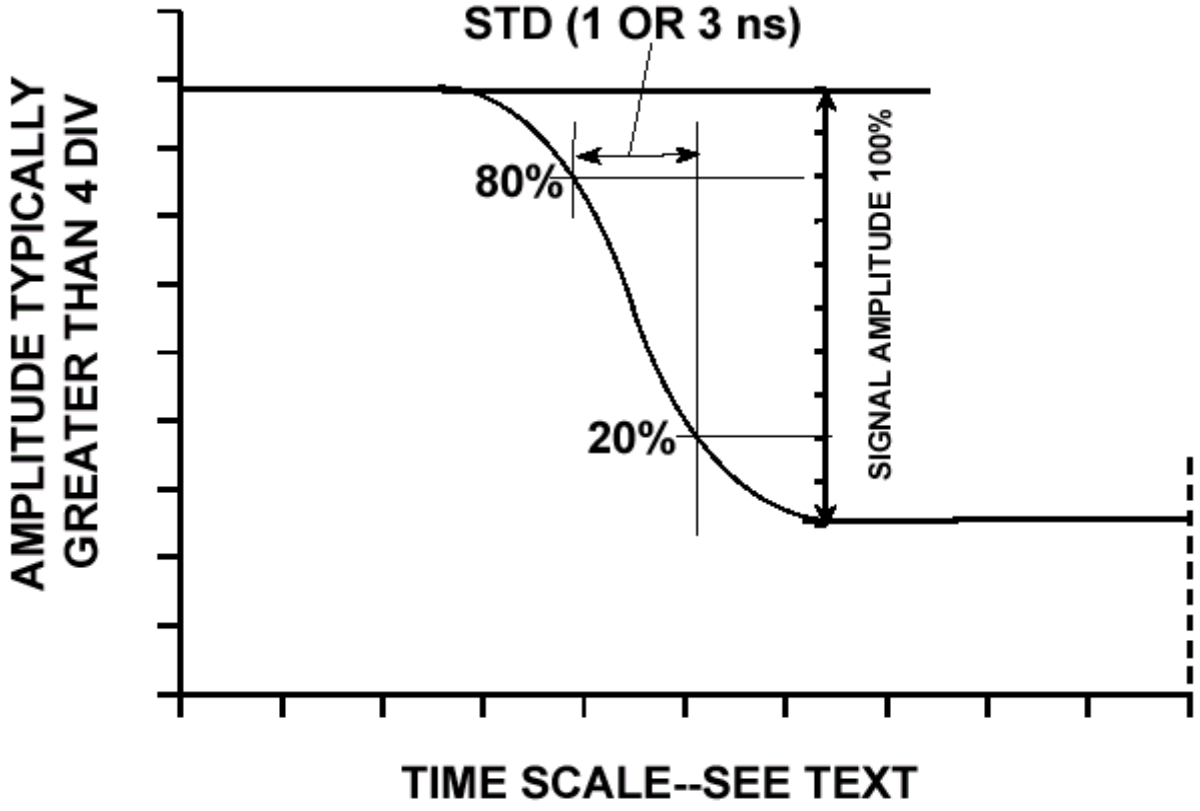


Figure E.3 - Signal transition duration calibration

E.2.1.5 Testing procedure

Connect the DUT to the test fixture wire clamp and record the TDR trace using the method described below.

a) Set the time scale to 2 ns / div (total time axis span of 20 ns).
b) Set the vertical scale (mρ) to 40 mρ / div.
c) With the DUT disconnected turn off the filtering.
d) Next, set the horizontal position such that the discontinuity is on the third division from the left.
e) Adjust the vertical position to approximately place the 50 Ω reference (cable from fixture to TDR) at the first vertical division from the bottom.
f) With the rise time filter adjusted to achieve 3 ns connect the DUT.
g) Unshielded DUT shall be suspended in air. No metallic supports should be used.
h) Set the TDR cursor to measure the minimum and maximum Ohms with cursors set on the trace as it crosses the 5^{th} and 6^{th} times divisions.

These measurements ignore the small error factor caused by losses in the cable which varies with gauge size. This error increases the measured impedance slightly.

Figure E.4 shows the TDR display setup to use for this measurement.

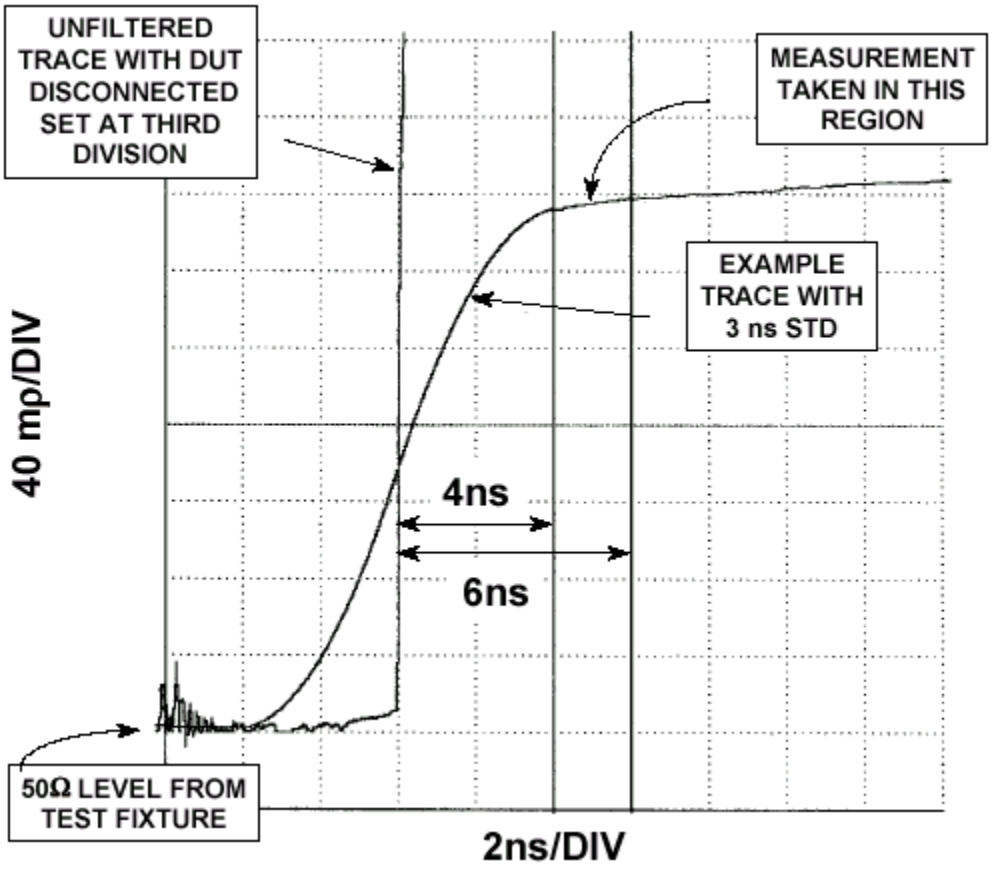


Figure E.4 - SE impedance measurement

E.2.2 Local impedance for differential transmission

E.2.2.1 Local impedance for differential transmission overview

This requirement is necessary to allow the cable media to interface with devices and terminators without inducing excessive signal reflections.

E.2.2.2 Sample preparation

This test requires type B samples (see table E.1) prepared in the following way:

1) Cut sample length to 3 ± 0,025 m.
2) Remove 5,0 cm of outer jacket from one end.
3) Comb out braid wire strands to form a pigtail.
4) Trim filler and tape materials to the base of braid wire.
5) Strip 0,5 cm insulation from all conductors.
6) Select the pair to be measured.

E.2.2.3 Test fixture and measurement equipment

Figure E.1 shows the test configuration for the local differential impedance test.

E.2.2.3.1 Calibration and verification procedure

E.2.2.3.2 Instrument verification

It is not necessary to perform a separate instrument verification for this test. The calibration in the following section includes the instrument.

E.2.2.3.3 Measurement system (with test fixture) calibration

Connect the 50 Ω cable to the test fixture.

In place of “B” in Figure E.1, connect a 137 Ω ± 0,1 % (preferred) low inductance chip resistor (i.e., IMS style TPI-1206 or equivalent). Use a differential unfiltered trace and use TDR cursors to measure the resistance value, R137, approximately 4 ns (displayed) after the resistor discontinuity. See figure E.2. The method shown in figure E.2 applies to differential except a 100 Ω level from the test fixture is seen and differential signals are displayed.

In a similar manner, in place of “B” in figure E.1, connect a 100 Ω ± 0,1 % (preferred) low inductance chip resistor (i.e., IMS style TPI-1206 or equivalent). Use a differential unfiltered trace and use the TDR cursors to measure the resistance value, R100, approximately 4 ns (displayed) after the resistor discontinuity.

For R137 and R100 the equation for determining the corrected (i.e., actual) impedance is:

$$Z_{corrected} = \frac{100 \times (((1,37 \times X_1) - X_2) - (0,37 \times Z_{mesasured}))}{X_1 - X_2}$$

Where:

a) X_1 is the measured value using the 100 Ω resistor
b) X_2 is the measured value using the 137Ω resistor

Figure E.5 shows an example of a differential calibration trace.

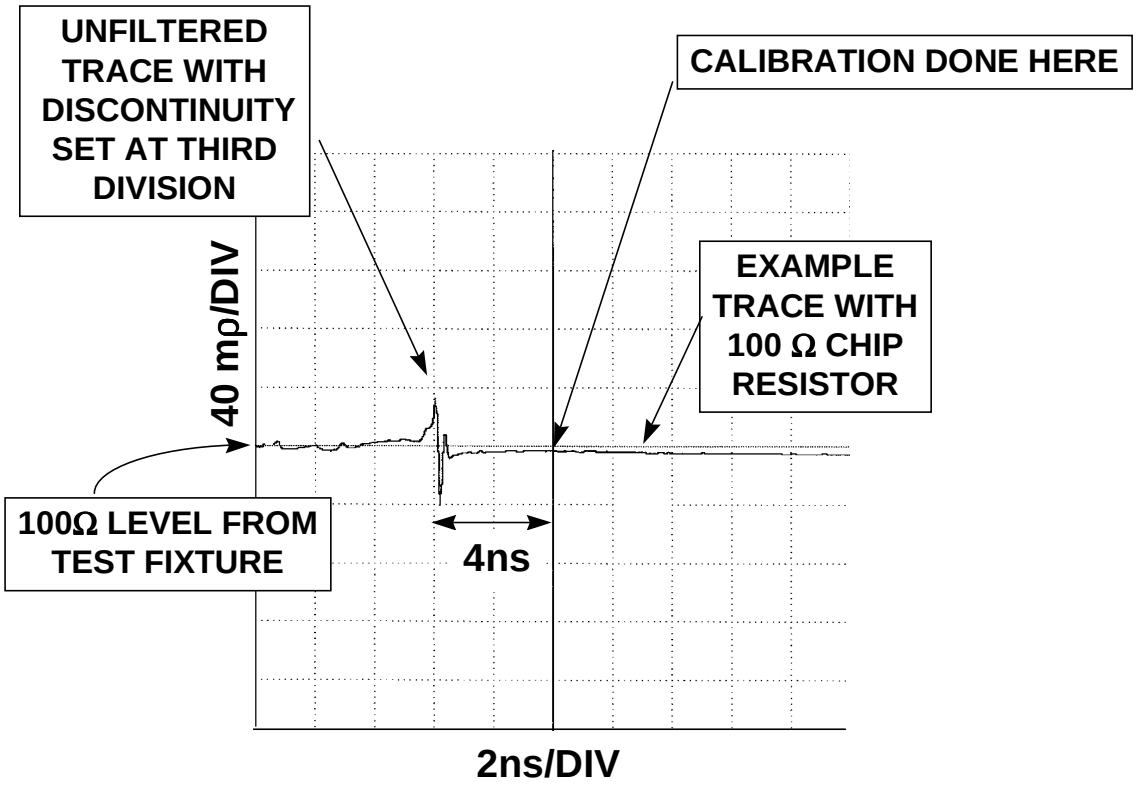


Figure E.5 - Example of a differential calibration trace

E.2.2.3.4 Differential signal transition duration (STD) calibration

This step ensures that the proper STD is being presented to the DUT. Place a short on the test fixture where the DUT would be attached (i.e., instead of “B” in Figure E.1). Use the filter function on the TDR to set the measured STD (20 % - 80 %) to the required value according to the detailed procedure described below. It may be desirable to use a separate test fixture that is nominally identical to the actual cable test fixture for this step.

Assuming a falling edge, set up the display on the TDR as shown in figure E.3. This display has the following properties:

a) The time scale on the display is 1,0 ns / div for the 1,0 ns STD and 2,0 ns / div for the 3,0 ns STD.
b) Set the horizontal position such that the midpoint of the displayed curve is near the center of the display and the 100 % and 0 % baselines are clearly visible as shown in figure E.3.

The STD is the time between the 20 % and 80 % values of the displayed signal amplitude (most instruments do this calculation automatically). When the instrument does not automatically measure STD, perform the following steps:

1) Measure the voltage at 100 % (V_{100}).
2) Measure the voltage at 0 % (V_0).
3) The voltage at 20 % is $V_{20} = V_0 + 0{,}2(V_{100} - V_0)$.
4) The voltage at 80 % is $V_{80} = V_0 + 0{,}8(V_{100} - V_0)$.
5) Set cursors at V_{20} and V_{80} and measure the time difference.

Adjust the TDR filter so that the required STD is achieved.

E.2.2.4 Testing procedure

Connect the DUT to the test fixture wire clamp and record the TDR trace using the method described below.

a) Set the time scale to 2 ns /div (total time axis span of 20 ns).
b) Set the vertical scale (mρ) to 40 mρ /div.
c) With the DUT disconnected turn off the filtering.
d) Next, set the horizontal position such that the discontinuity is on the third division from the left.
e) Adjust the vertical position to approximately place the 100 Ω reference (cable from fixture to TDR) at the fifth vertical division from the bottom.
f) With the rise time filter adjusted to achieve 3 ns connect the DUT.
g) Unshielded DUT shall be suspended in air. No metallic supports should be used.
h) Set the TDR cursor to measure minimum, mean, and maximum Ohms with cursors set on the trace as it crosses the 5^{th} and 6^{th} times divisions.
i) Set the filter to 3 ns.
j) Set the TDR cursor to measure minimum and maximum Ohms with cursors set on the trace as it crosses the 5^{th} and 6^{th} times divisions.

Figure E.6 shows the TDR display setup to use for this measurement.

These measurements ignore the small error factor caused by losses in the cable which varies with gauge size. This error increases the measured impedance slightly.

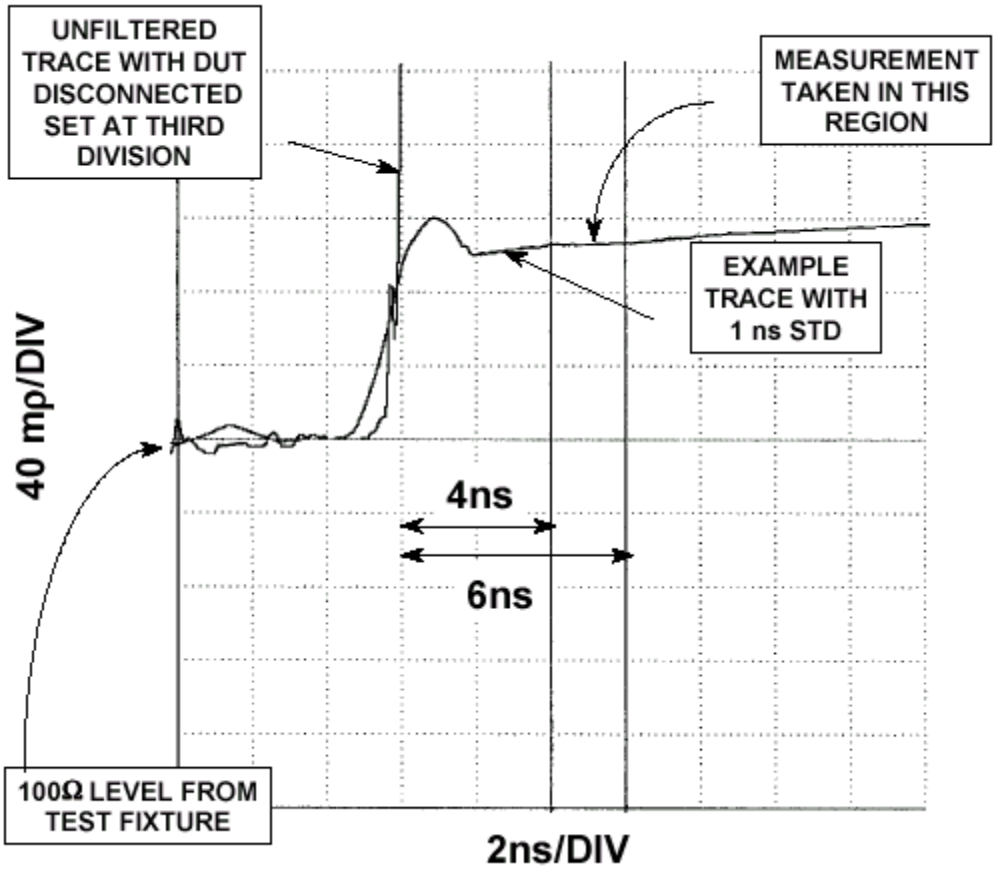


Figure E.6 - Differential impedance test

E.2.3 Differential extended distance (balanced) impedance (frequency domain)

E.2.3.1 Sample preparation

This test requires type C samples (see table E.1) prepared in the following way:

1) First the cable sample shall be cut to a length such that resonance does not occur. (Approximately 30 m or greater.)
2) Remove 5,0 cm of outer jacket from both ends.
3) Comb out braid wire strands to form a pigtail.
4) Trim filler and tape materials to the base of braid wire.
5) Strip 0,5 cm insulation from all conductors.
6) Attach a 122 Ω resistor to the far end and across the pair under test.

E.2.3.2 Test fixture for differential extended distance impedance

E.2.3.2.1 Test fixture for differential extended distance impedance figure

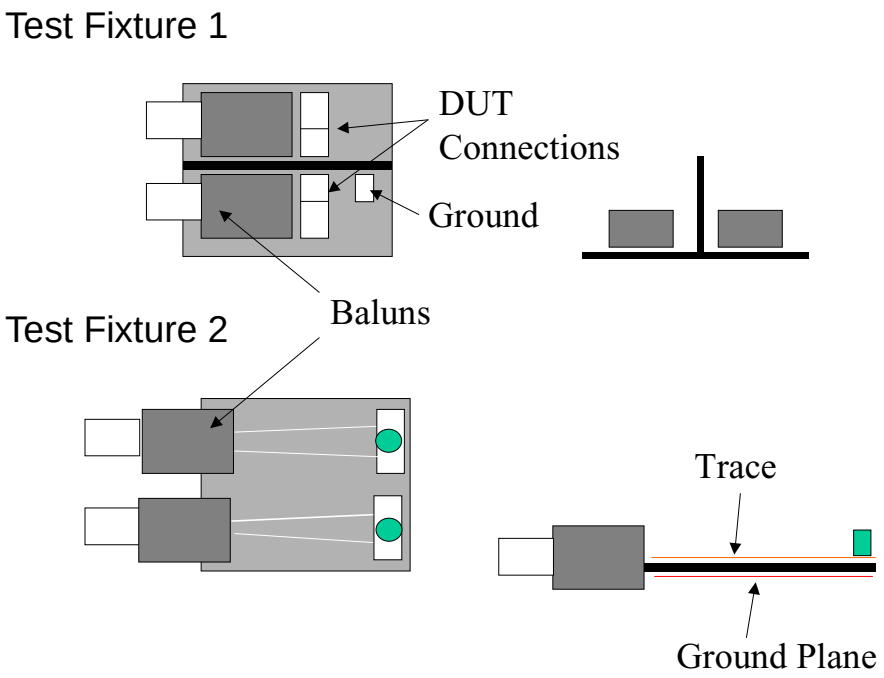


Figure E.7 - Test fixtures for differential extended distance impedance profile

Figure E.7 shows the test configuration for the differential extended distance impedance tests.

E.2.3.2.2 Scope and objective

Two types of test fixtures are described in order to provide sufficient flexibility in constructing fixtures required in measuring cable characteristics in the frequency domain environment. The two types of fixtures are closely related.

The first type is constructed using two baluns mounted on metallic base and electrically isolated from each other by metallic screen **(**test fixture 1 figure E.7).

The second type uses high speed PCB with microstrip construction. Two baluns are mounted at one edge

of the board with sufficient separation to reduce mutual coupling by a minimum of 20 dB. The interconnect traces of the signal pairs are further separated from each other in a radial form and the signal traces run at 61 Ω nominal to ground for each differential line (test fixture 2 figure E.7).

E.2.3.2.3 Test equipment

A scalar or vector network analyzer instrument measures the reflection and transmission characteristics of the DUT. With the appropriate calibration and error correction techniques, a high degree of accuracy is possible. The analyzer may be used both as the source of the test signal and as a means of measuring the DUT's characteristics.

Two precision coax cables whose transmission line impedance matches the impedance of the network analyzer system are used to connect the test fixture to the test analyzer system.

E.2.3.2.4 Test fixtures

For the metallic base test fixture (test fixture 1), the transmission line is provided by the baluns as the signal paths and the metallic base as the current return path. The baluns provide a 50 to 61 Ω impedance matching between the test instrument system and the DUT, as well as provide differential signals. The metallic screen provides electric field isolation between the two baluns. The DUT connects to the fixture via a mechanical clamp system. The DUT should self terminate given its length.

E.2.3.2.5 Fixture board design requirements (test fixture 2 figure E.7)

For the PCB test fixture, the test fixture consist of a printed circuit board incorporating controlled impedance trace construction of 61 Ω (refer to test fixture 2 figure E.7). The transmission line is provided by the connected baluns and PCB traces for the signal paths and the ground plane of the board for the return current path. A coaxial cable (same transmission line impedance as the test instrument) connects one end of the cable to the instrument test port through the baluns and the PCB fixture trace combination. The baluns provide a 50 to 61 Ω impedance matching between the test instrument system and the DUT, as well as provide differential signals. The separation of the baluns and the signal lines provide electric field isolation between the two baluns and the signal lines. The DUT connects to the fixture via a mechanical clamp system. The DUT should self terminate given its length.

NOTE 61 - The baluns required for this test are high frequency (650 MHz or greater) precision types.

A stand is recommended for mounting fixture board and to support the DUT. It is recommended that the stand keep the fixture board at least 7 cm from the top of the lab bench to minimize coupling.

Traces are constructed on the PCB to conform with the differential transmission scheme. The fixture shall be through hole or surface mount PCB. The signal traces are connected to the balun's differential pins using microstrip construction with controlled characteristic impedance of 61 Ω. The length of the connections shall permit the board to operate at the required frequencies and accommodate the required number of signal lines, including sufficient separation to reduce mutual coupling by a minimum of 20 dB. It is recommended that the bandwidth of the board be at least 650 MHz. Board impedance shall be controlled to within 5 % of the nominal impedance.

The fixture board shall include calibration traces for measuring the effects of the test fixturing on the measurement data. This board construction is useful for other frequency domain measurements but is not designed to accommodate time domain.

E.2.3.3 Calibration procedure

Set the analyzer to perform an S_{11} (see notes in table E.1) measurement with the source power set at a minimum of 6 dbm, the number of points set to a minimum of 401 and the band width at a maximum of 200 Hz (averaging at a minimum of 2 averages). Start and stop frequencies are 1 MHz and 1 GHz respectively. The calibration is for the open, short and load method, keeping the leads as short as possible on the load

standards. The load standard shall match the impedance of the secondary balun and have a maximum tolerance of 1 %.

E.2.3.4 Measurement procedure

Using either:

a) Hewlett Packard 4291B RF Impedance / Material Analyzer or equivalent. Test fixture and an impedance matching balun unbalanced to balanced, or
b) Hewlett Packard 8753E Network Analyzer or equivalent test fixture and an impedance matching balun unbalanced to balanced.

Unshielded DUT shall be suspended in air. No metallic supports should be used.

Connect the near end of the sample to the output balun on the test fixture, keeping the leads as short as possible to eliminate inductance problems and terminate the far end of the sample in the impedance of the cable. Perform a sweep test from 1 MHz through 1 GHz.

E.3 Capacitance

E.3.1 Single ended capacitance

E.3.1.1 Sample preparation

This test requires type D samples (see table E.1) prepared in the following way:

For planar cables:

1) Cut sample length to 3 m.
2) Separate conductors at one end.
3) Strip 0,5 cm insulation from all conductors.

For round cables shield connected:

1) Cut sample length to 3 m.
2) Remove 5,0 cm of outer jacket from one end.
3) Comb out braid wire strands to form a pig tail.
4) Trim filler and tape materials to the base of braid wire.
5) Strip 0,5 cm insulation from all conductors.
6) Connect one (1) conductor of each pair to the shield.

E.3.1.2 Test fixture for single ended capacitance

Figure E.8 shows the test configuration for the single ended capacitance tests.

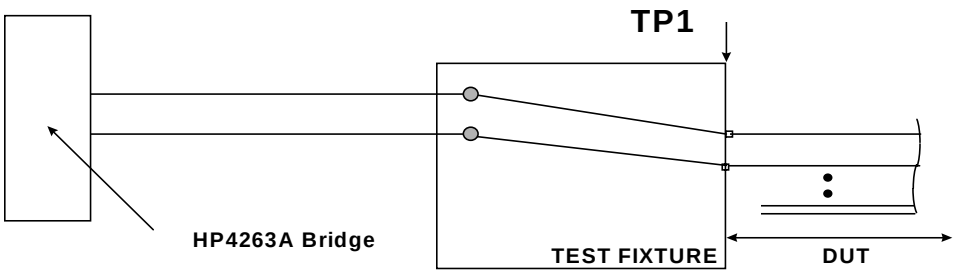


Figure E.8 - Test fixture for single ended capacitance measurement

E.3.1.3 Calibration procedure

If using an HP4263A or equivalent bridge, calibration shall be as follows:

1) Connect fixture to meter and perform open calibration as specified by the bridge.
2) Connect a wire (short) to the sockets of the test fixture and perform a “short” calibration as specified by the bridge.

For other manufacture’s equipment, follow the calibration procedures specified by the manufacturer of the LCR bridge for reliable results.

E.3.1.4 Measurement procedure

E.3.1.4.1 Planar cables - G-S-G

With the bridge set at the required frequency, connect the two grounds together on one side of the test fixture and the signal to the other side of the test fixture, record the capacitance.

E.3.1.4.2 Round cables - shielded

With the bridge set at the required frequency, connect one conductor of the twisted pair to one side of the test fixture. Connect the common conductors and shield to the other side of the test fixture (ground). Record the capacitance.

E.3.2 Differential capacitance

E.3.2.1 Sample preparation

This test requires type F sample (see table E.1) prepared in the following way:

For planar cables:

1) Cut sample to a length that eliminates resonance.
2) Separate conductors at both ends.
3) Strip 0,5 cm insulation from all conductors.

For round cables shield floating:

1) Cut sample to a length that eliminates resonance.
2) Remove 5,0 cm of outer jacket from both ends.

3) Cut braid wire back to jacket.
4) Trim filler and tape materials to the base of braid wire.
5) Strip 0,5 cm insulation from all conductors.

E.3.2.2 Test fixture for differential capacitance

Figure E.9 shows the test configuration for the differential capacitance tests.

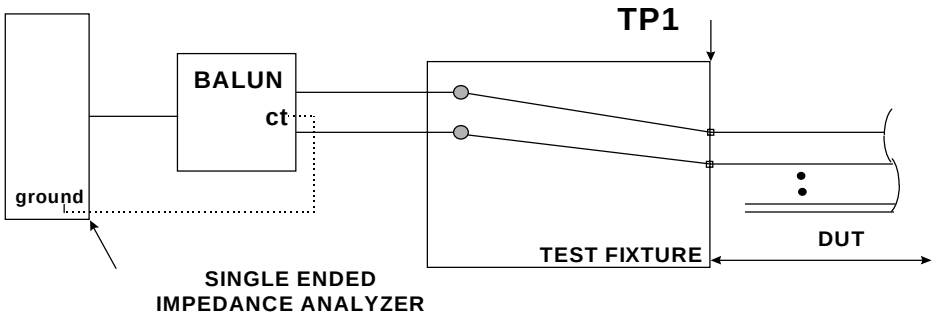


Figure E.9 - Test fixture for differential capacitance measurement

E.3.2.3 Calibration procedure

E.3.2.3.1 Calibration procedure equipment

If using an impedance/material Analyzer Model #4291B from HP or equivalent, calibration shall be as follows:

E.3.2.3.2 Calibration set up procedure

1) Set up bandwidth.
2) Set start and stop frequencies.
3) Set for impedance magnitude for measurement type.
4) Number of points is equal to 401.
5) Minimum average points shall be two (2).
6) Record the linear measurement.

E.3.2.3.3 Calibration

1) Connect fixture to test head and perform open circuit calibration.
2) Connect wire (short) to test fixture head and perform “short” circuit calibration.
3) Connect a 50 Ω resistor to the test fixture and perform load calibration.

For other instruments, follow the calibration procedures specified by the manufacturer for reliable results.

E.3.2.4 Measurement procedure

The measurement shall be performed using the equivalent circuit method adjusting the open and short values of the capacitance and inductance so that the traces over lay as closely as absolutely possible across the entire bandwidth.

Impedance values could be measured and recorded at this time.

NOTE 62 - Fixture should be measured for functionality through bandwidth of interest.

E.4 Propagation time (differential frequency mode)

E.4.1 Sample preparation

This test requires type G samples (see table E.1) prepared in the following way:

1) Cut sample length to 30 ± 0,1 m.
2) Remove 5,0 cm of outer jacket from both ends.
3) Comb out braid wire strands to form a pigtail at both ends.
4) Trim filler and tape materials to base of braid wire at both ends.
5) Strip 0,5 cm insulation from all conductors.

E.4.2 Test fixture for propagation time (differential frequency mode)

See figure E.18 for appropriate test fixtures.

E.4.3 Calibration procedure

The analyzer shall be set to perform an S_{12} measurement with the power set at a minimum of 6 dbm, the number of points set to a minimum of 401, the resolution bandwidth at a maximum of 200 Hz, averaging at a minimum of 2 averages, and the start/stop frequencies per table E.1. Perform a transmission calibration using a sample of the cable to be tested keeping the sample as short as possible.

E.4.4 Measurement procedure

With the analyzer set up in the delay mode, connect one end of the sample to the balun on the output port and the opposite end to the balun on the input port. With the markers turned on record the minimum and maximum delay across the bandwidth as listed in table E.1.

E.5 Propagation time (differential - time domain mode)

E.5.1 Sample preparation

This test requires type H sample (see table E.1) prepared in the following way:

For planar cables:

1) Cut sample length to 6 m.
2) Separate conductors at one end.
3) Strip 0,5 cm insulation from all conductors.

For round cables shield floating:

1) Cut sample length to 6 m.
2) Remove 5,0 cm of outer jacket from one end.
3) Comb out braid wire strands to form a pig tail.
4) Trim filler and tape materials to the base of braid wire.
5) Strip 0,5 cm insulation from all conductors.

E.5.2 Test fixture for propagation time (differential - time domain mode)

Figure E.10 shows the test configuration for the propagation time tests.

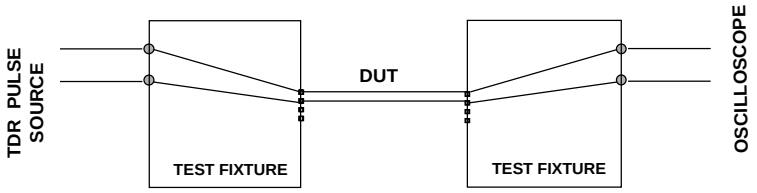


Figure E.10 - Test fixture for propagation time measurement

E.5.3 Test configuration calibration procedure

If using an TDR scope such as an HP54750 or equivalent, calibration shall be as follows:

1) Leave channel 1 and 2 open;
2) Default setup;
3) Setup, stimulus, differential;
4) Preset TDR/TDT;
5) Select time base - set to 100 ps/div;
6) Select position - move trace to approximately center of screen;
7) Select acquisition - change average to 4;
8) Delta time - Channel 1 minus Channel 2;
9) Connect test fixture;
10) Press time base - select position;
11) Adjust position of fixture;
12) Select right most channel;
13) Select calibrate;
14) Select skew;
15) Reduce imbalance by half;
16) Select TDR skew - adjust until difference between the two channels is approximately zero;
17) Clear markers;
18) Select TDR/TDT setup;
19) Establish reference plane;
20) Calibrate channel one;
    A) Connect a wire (short) to the sockets of the test fixture and perform a “short” calibration as specified by the TDR scope.
    B) Connect a 50 Ω precision resistor to the test fixture and perform load calibration.
21) Calibrate channel two;
    A) Connect a wire (short) to the sockets of the test fixture and perform a “short” calibration as specified by TDR scope.
    B) Connect a 50 Ω precision resistor to the test fixture and perform load calibration.

For other instruments, use the same procedure adapted for that instrument.

At this point calibration is complete. To setup for the measurement the following additional steps are required:

1) Select TDR response 1
2) Select differential
3) Vertical resolution shall be set to display the full dynamic range of the response.
4) Turn off channel 1
5) Turn off channel 2

6) The differential response is displayed
7) Markers
   A) Set + source to response 1
   B) Set x source to response 1
8) Adjust markers so they are on the trace

E.5.4 Measurement procedure

The measurement procedure is as follows:

1) Select TDR response 1;
2) Select differential;
3) Turn off channel 1;
4) Turn off channel 2;
5) The differential response is displayed;
6) Adjust both markers so that they track the response trace;
7) Connect the DUT to the test fixture;
8) Vertical resolution shall be set to display the full dynamic range of the response;
9) Adjust time base scale so that both the launch and the end of cable are on the screen;
10) Set the + marker to the temporal position at the midpoint of the launch step (signal transition);
11) Set the x marker to the temporal position at the midpoint of the end step;
12) Set horizontal unit to seconds;
13) Take the time delta between markers and divide by twice the cable length. This provides the propagation time.

E.6 Propagation time skew (differential)

E.6.1 Measurement procedure

Using the time domain (through) measurement, maximum propagation time minus the minimum propagation time renders the overall propagation time skew of the pair under test.

E.7 Attenuation

E.7.1 Differential attenuation

Attenuation is calculated from the ratio of output signal level to input signal level through the DUT and is a measure of the losses experienced when transmitting a signal through the DUT. Higher attenuation means less signal at the output or equivalently a gain of less than unity. A sinusoidal signal is used to eliminate the need for complex descriptions of real pulses and square or trapezoidal signals in terms of Fourier components. A complete attenuation specification requires examining all frequencies of interest to the application. A spectral description is recommended. The basic formula for attenuation in decibels is:

$$\text{Attenuation (dB)} = 20 \log_{10} (\text{input voltage} / \text{output voltage}).$$

This formula gives the attenuation as a positive number since the argument of the log is greater than unity. Sometimes attenuation is casually reported as a negative number when the gain is really the intended mathematical statement. In any case the magnitude is the same for both gain and attenuation. The following formula expresses gain in decibels.

$$\text{Gain (dB)} = 20 \log_{10} (\text{output voltage} / \text{input voltage})$$

Since the argument of the log is less than unity the gain is a negative number.

This standard requires that attenuation be expressed as a positive number unless there is active gain in

the path from active circuits. Therefore a typical attenuation plot has the form shown in figure E.11.

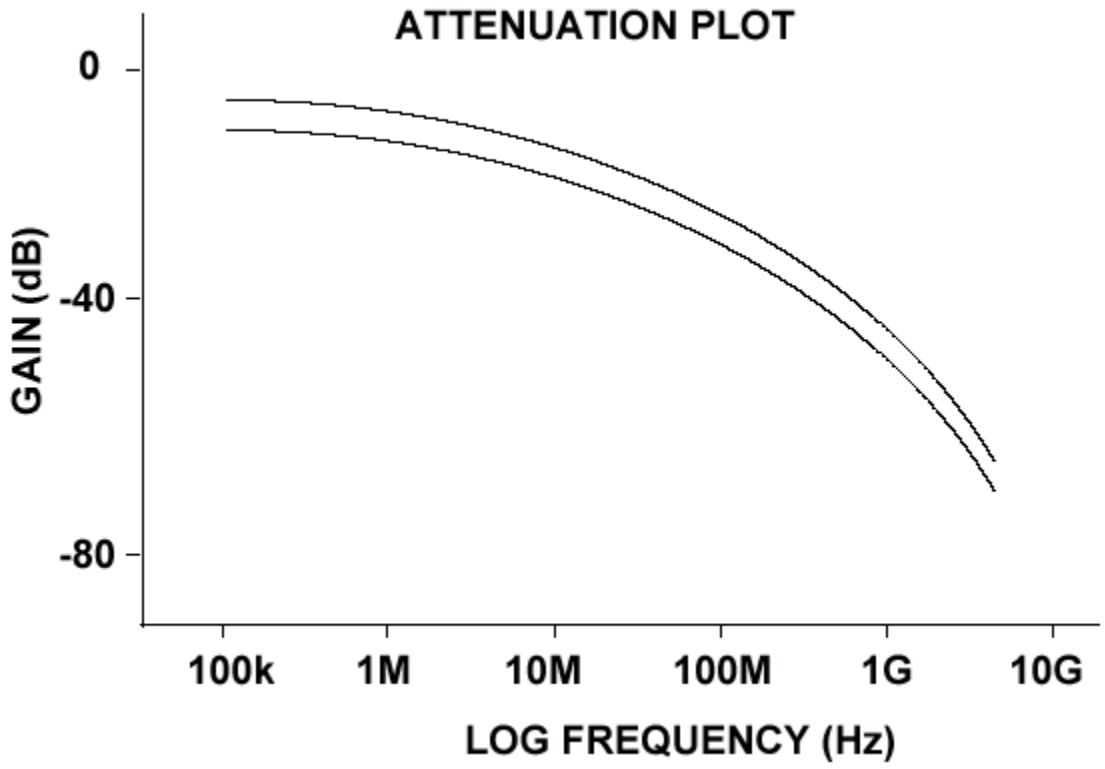


Figure E.11 - Form of attenuation plots

Attenuation is a measurement of the dissipative losses on a balanced transmission line. The series resistive loss of the conductors (copper) and the shunt loss due to the dissipation factor of the dielectric covering the conductors dominate these losses. At higher frequencies, the conductor loss increases due to skin effect. Skin effect is where the current becomes increasingly confined to the outer “skin” of the conductor as the frequency increases. This effectively reduces the conductor area available for current flow. The attenuation for a given transmission line will be affected by the conductor metal composition and size, and the composition, uniformity, and thickness of the dielectric that surrounds the conductors.

Attenuation is only measured directly with an ideal test system that is perfectly matched to the balanced transmission line to be tested. In a practical test system, the quantity that is actually measured is insertion loss. Insertion loss is comprised of a component due to the attenuation of the balanced transmission line, a component due to the mismatch loss at the input or near end side of the transmission line and a component due to the mismatch loss at the output or far end side of the transmission line.

There is a mismatch loss component at any interface where the transmission line impedance is not perfectly matched on both sides of the interface. The amount of mismatch loss that is experienced at each interface is:

$$\text{Mismatch Loss (dB)} = (-\ 10\ \text{LOG}_{10}\ (1 - |\ \Gamma\ |^2\ ))\ \text{dB}$$

Where:

Γ = reflection coefficient

Balanced transmission lines are also susceptible to measurement errors when measuring high values of

attenuation (>50 dB) due to radiated energy coupling into the transmission line. The largest source of this error is due to direct coupling of the near end side of the test system to the far end side of the test system. This coupled signal combines with the test signal passing through the transmission line under test and cause a significant ripple error in the insertion loss measurements at the higher frequencies where the attenuation of the transmission line under test is the largest.

Since the test instruments are single ended and the DUT is balanced a coupling device called a balun is required to connect the DUT to the test equipment.

E.7.2 Sample preparation

1) Cut sample to a length that produces at least 1 dB attenuation at the low frequency shelf (typically at least 30 meters).
2) Remove 5,0 cm of outer jacket from both ends.
3) Cut braid wire back to jacket.
4) Trim filler and tape materials to the base of braid wire.
5) Strip 0,5 cm insulation from all conductors.

E.7.3 Balun selection

The impedance on the primary side of the balun shall match the impedance of the test equipment, normally a network analyzer. The impedance on the secondary side of the balun shall be matched as closely as possible to the nominal impedance of the DUT in the balanced state to minimize reflections. If reflections are present they skew the data by introducing a mismatch loss ripple component.

Figure E.12 shows the effect of different baluns on a very long cable (approximately 300 m). There are no reflections visible because they are attenuated to insignificant levels by the long length. Another very important benefit to using long cables for these tests is the elimination of resonance effects for the same reason that reflections are not a problem. The main effect of using relatively serious mismatched baluns on very long cables is a small error in the attenuation reported (less than 1 dB in the example shown).

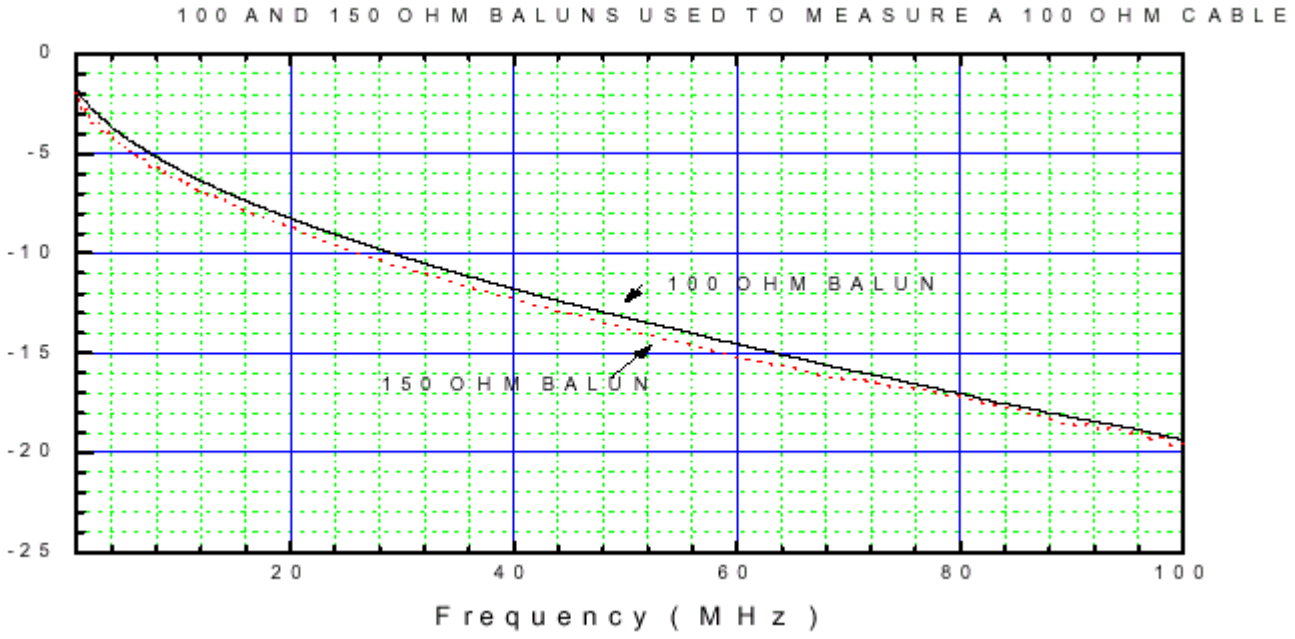


Figure E.12 - Effect of balun selection on measured attenuation for very long cables

E.7.4 Sample length

The optimum sample length is such that there is approximately 1 dB of one way attenuation at the lowest frequency of interest. This guarantees that there is at least 2 dB of additional loss experienced by that portion of the test signal that reflects from the far end.

This reduces the uncertainty caused by multiple reflections due to the far end and results in acceptable resolution / ripple. The resulting measurements are then accurate and repeatable. If a sample is used that yields an attenuation of less than 1 dB the mismatch ripple from the near end combined with the mismatch ripple from the far end may approach the same or greater magnitude than the attenuation at the lowest frequency.

An example of an unmatched balun case follows:

Near End Balun Z = 100 Ω
Far End Balun Z = 100 Ω
Nominal Balanced Cable Z = 150 Ω
Balanced Cable loss at lowest test frequency = 0,5 dB = 0,94406

In this case there is only 0,5 dB attenuation in the presence of a ripple that adds 0,354 dB to 0,6972 dB of measurement error as shown in figure E.13.

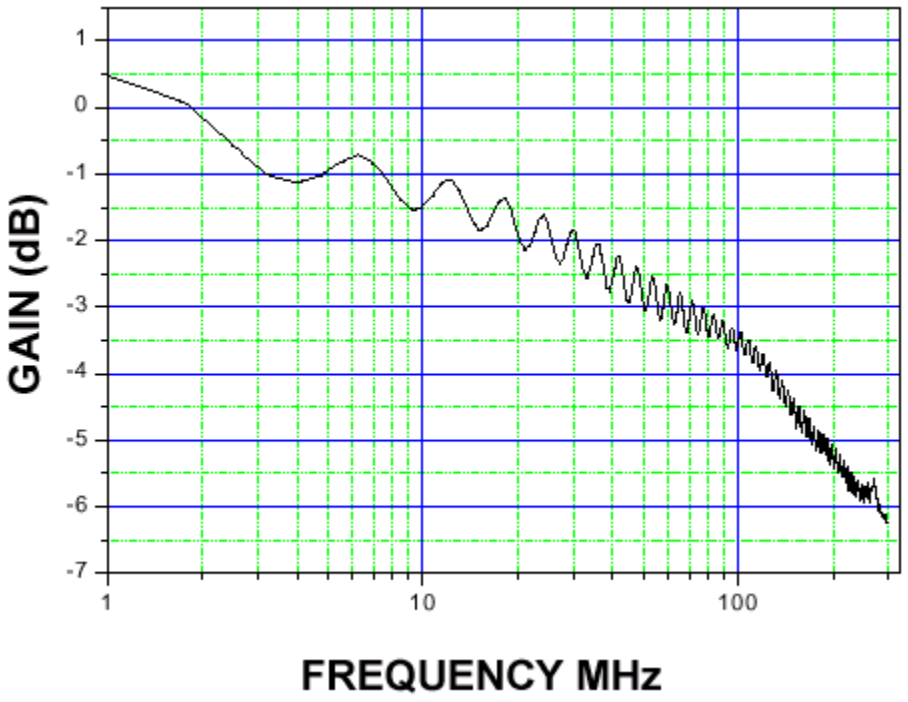


Figure E.13 - Effects of mismatched baluns in a short DUT

The next example is a matched case otherwise identical to unmatched case above.

An example of a matched balun case follows:

Near End Balun Z = 150 Ω
Far End Balun Z = 150 Ω
Nominal Balanced Cable Z = 150 Ω
Balanced Cable loss at lowest test frequency = 0,5 dB = 0,94406

The insertion loss equals the required attenuation result and there is no ripple to cause measurement uncertainty as shown in figure E.14.

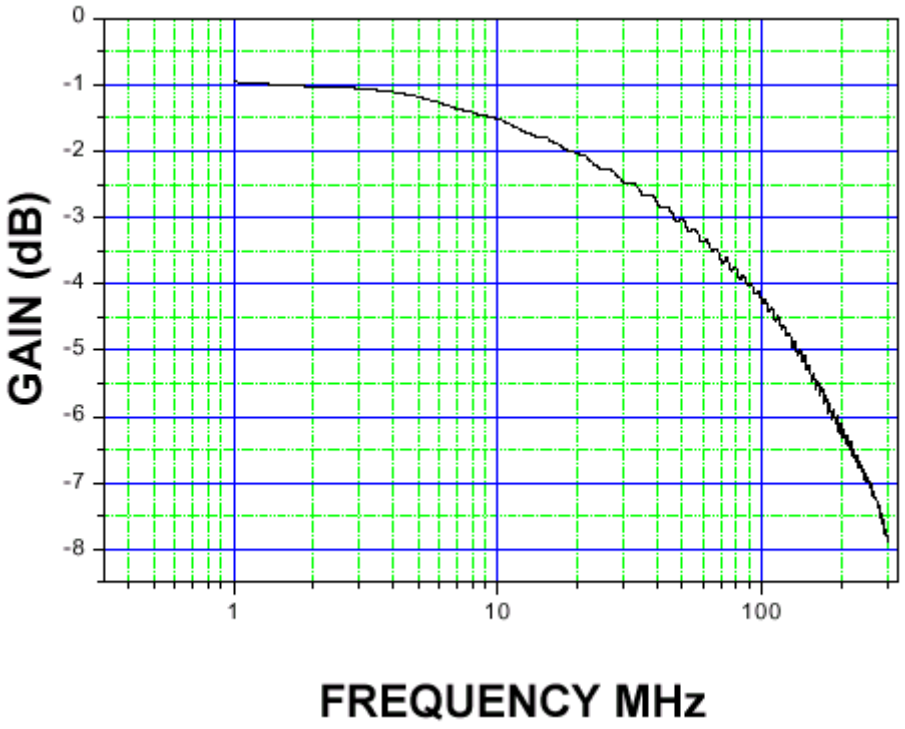


Figure E.14 - Effect of matched baluns on a short sample

For the case of at least 6 dB of low frequency attenuation the results are achieved without requiring a closely matched balun as shown in the example below.

Near End Balun Z = 100 Ω
Far End Balun Z = 100 Ω
Nominal Balanced Cable Z = 150 Ω
Balanced Cable loss at lowest test frequency approximately 6 dB

Under a mismatched condition, the insertion loss equals the attenuation plus the mismatch loss at the near end and at the far end. However, in this case, there is sufficient attenuation in the cable at the lowest frequency to make multiple reflections inconsequential, so there is no ripple component of measurement uncertainty. The mismatch loss error is still present, but it is approximately 0,3 dB out of a measured insertion loss of approximately 6 dB.

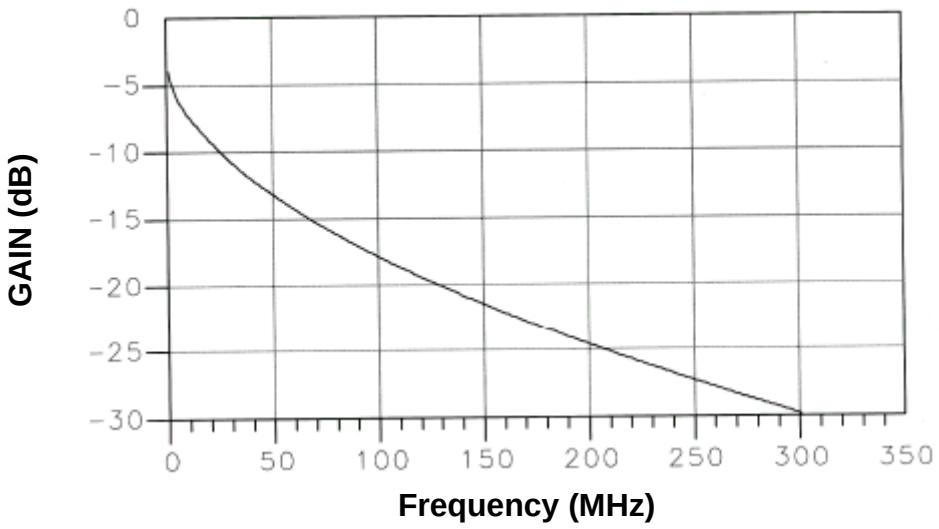


Figure E.15 - Effects of mismatched baluns with 6 dB LF attenuation

The use of an attenuator for measuring shorter lengths is not acceptable because there is still mismatch loss uncertainty due to the fact that the attenuator does not have any better match than the far end test port. Also an additional uncertainty is introduced because a small value (cable attenuation) is being subtracted from a large value (attenuator attenuation). There are also dynamic range issues for the instrumentation.

E.7.5 Measurement test fixture and measurement equipment

An instrument capable of supplying a sinusoidal signal is used as the signal source and an instrument capable of detecting the amplitude of a sinusoidal signal is used as the signal sink. Two measurement test fixtures are required: one for the source end and one for the sink end. Since most source and sink instruments capable of using variable frequency sinusoidal signals are single ended, a balun (e.g., Picosecond Pulse Labs) or a hybrid (e.g., Picosecond Pulse Labs, Minicircuits) may be used between the instruments and the test fixtures. If a source or sink is used that is capable of sourcing or sinking differential signals then no balun is required for the differential source or sink.

Equipment Required: Network Analyzer (HP 87xx Series) or equivalent.

A test fixture having 61 Ω single ended paths for each signal line is used for the measurement as shown in figure E.16 and figure E.17 and calibrated as shown in figure E.18. This test fixture may be exactly the same as used for the impedance tests in E.2.1.

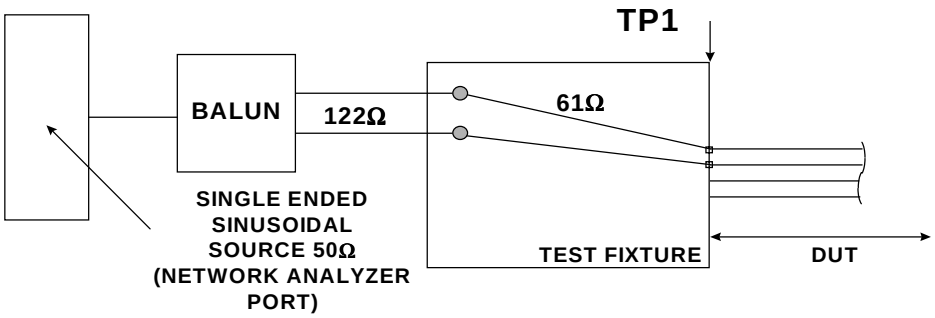


Figure E.16 - Source-end test fixture for attenuation tests

The balun shown in figure E.16 is 50 Ω single ended to 122 Ω differential.

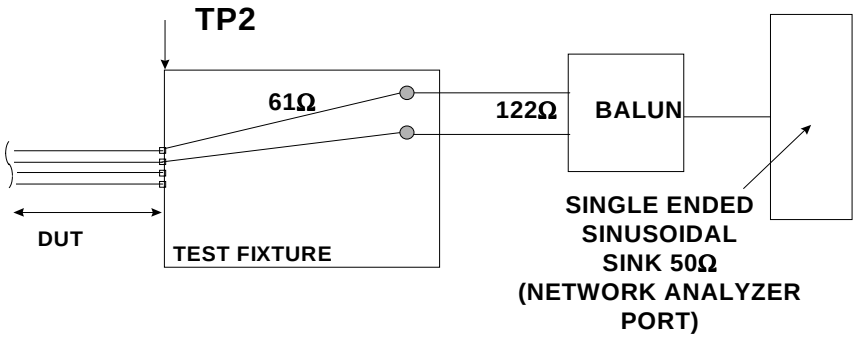


Figure E.17 - Sink end test fixture for attenuation tests

The DUT is connected between the source and sink test fixtures.

E.7.6 Calibration procedure

A special “through” test fixture is used for the signal calibration process which is exactly like the test fixture in figure E.16 and figure E.17 except that there are no DUT connectors (A, D). See figure E.18.

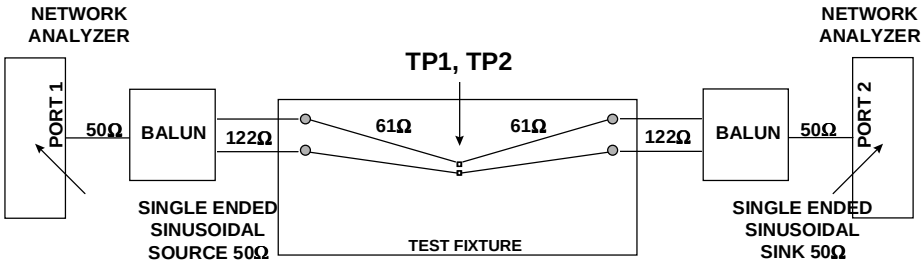


Figure E.18 - Calibration configuration for attenuation tests

Using the instructions for the network analyzer perform a calibration scan over the frequency of interest.

E.7.7 Testing procedure

Connect the DUT to the test fixtures shown in figure E.16 and figure E.17 (including the board mounted connectors).

Using the instructions from the network analyzer perform an attenuation scan over the frequency range of interest. The instrument should automatically account for the attenuation found in the calibration scan.

It is important to either separate or shield the baluns from each other when measuring long cable samples. When the attenuation of the cable exceeds approximately 50 dB or the frequency is above approximately 150 MHz, potential direct coupling from the near end to the far end balun creates an increasingly large ripple in the attenuation measurement that may cause a significant amount of measurement uncertainty.

Figure E.19 and figure E.20 show the effects of balun isolation.

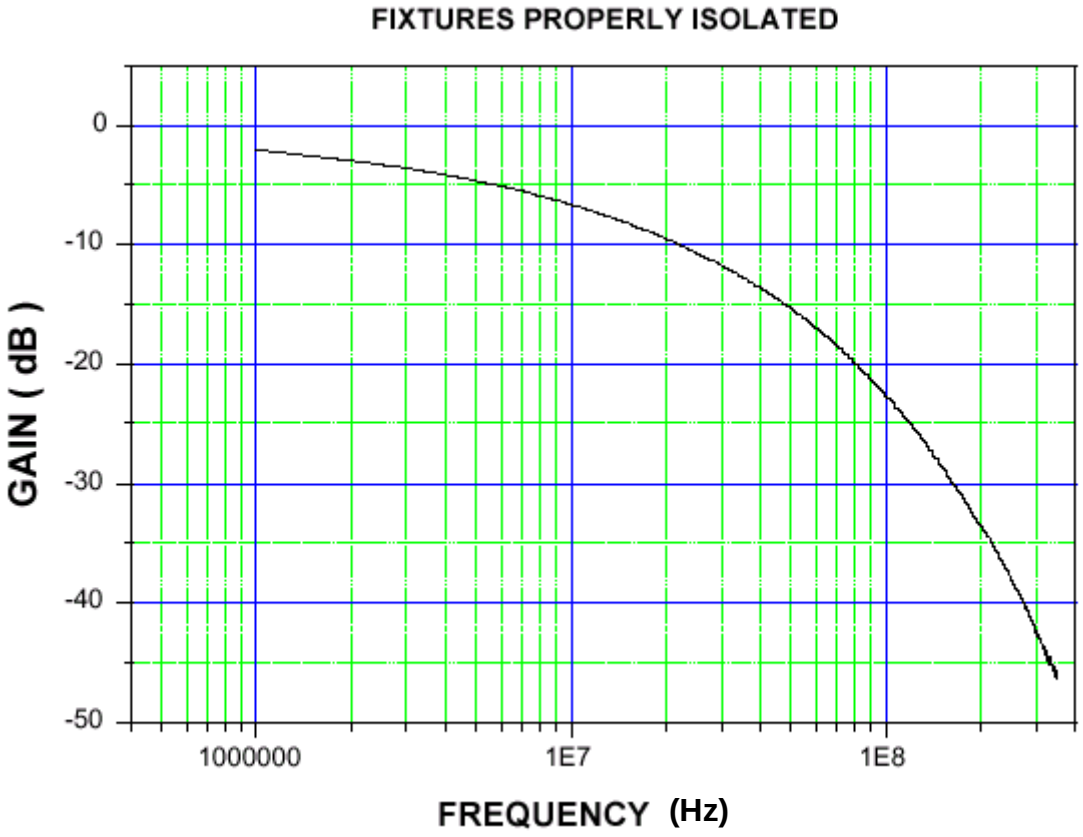


Figure E.19 - Attenuation scan with proper balun isolation

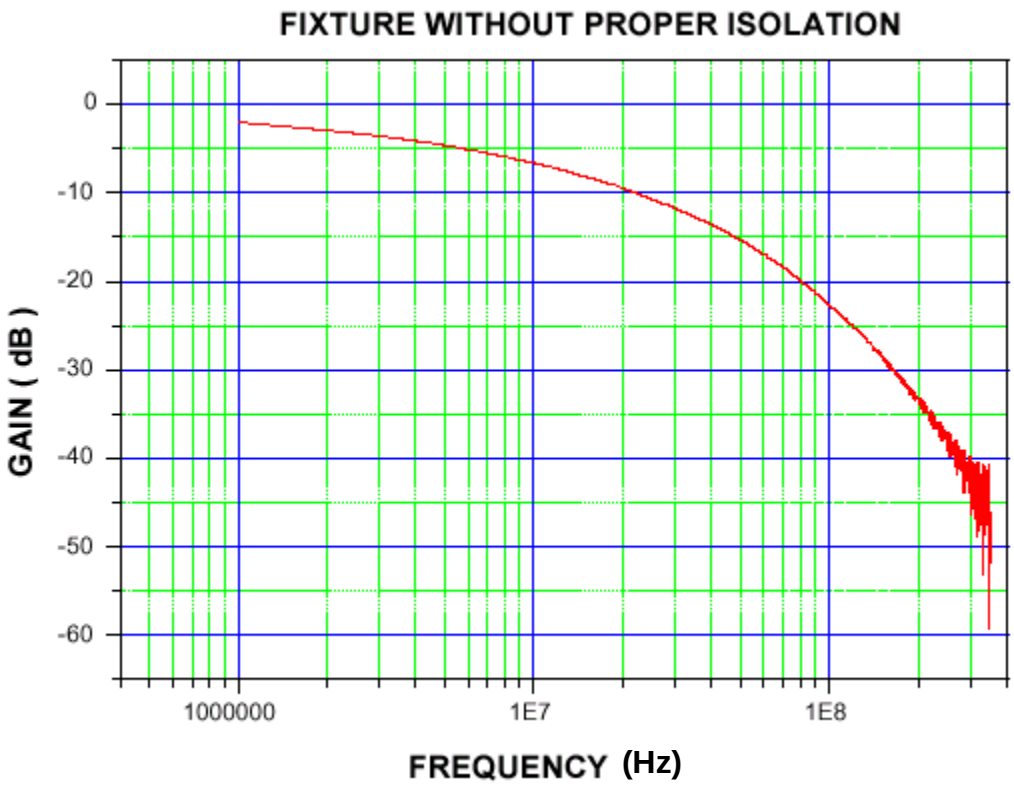


Figure E.20 - Attenuation scan without good balun isolation

E.8 Near end crosstalk (quiescent noise)

E.8.1 Near end crosstalk description

This test is limited to the single applied pulse method. The sum of the noise produced by all pairs on the REQ or ACK pair is the crosstalk. In this method pulses with maximum differential amplitude, maximum and minimum STD signal are applied to one pair at a time in the DUT, and the signal induced on the REQ or ACK pair is measured. The pair with the applied pulse is the aggressor pair and a pair with the induced noise is the victim pair. If the REQ is the victim pair then the ACK pair is included as an aggressor pair. If ACK is the victim pair then the REQ pair is included as an aggressor pair. The sum of the noise from the aggressor pairs on the victim pair is the crosstalk.

Single pulse tests that eliminate the effects of resonance, are very deterministic in the causes of the induced noise (due to the mapping of the time and space as in the TDR tests), and produce the worst case results. It is necessary to reverse the polarity of the aggressor signal to ensure that unintended compensation from the imperfections in the applied signal is not occurring.

The aggressor pulses are of the same type used for the impedance test: start with single ended signals: + signal at ± 250 mV and the - signal at ± 250 mV. The + signal and - signal pulses initiate in opposite directions to form a collapsing differential aggressor pulse ending at differential zero.

The use of actual worst case data patterns on the aggressor lines has been extensively debated and considered. This is the natural excitation that is initially considered. Extensive testing has shown that resonance conditions and effects of test fixtures may severely distort the measured results when using real data patterns. Sometimes these effects improve the crosstalk performance and other times they exacerbate it. It is very difficult to diagnose the intensity and cause of resonance and fixture effects when

using a real data pattern. The single pulse eliminates these effects and gives a worst case result that may be attributed to as much of the system as required. For example, if connector termination techniques are causing the crosstalk then that may be revealed by examining the time points associated with the termination points.

The value of the recorded disturbance in the victim line is the differential peak value of the induced noise at a time position within the DUT.

This requirement may appear contrary to logic that says the maximum disturbance occurs with the maximum signal swing that occurs with a peak to peak measurement. The reason that the peak measurement is the important parameter is that receivers measure the differential signal from a differential zero position. Even if the intensity of the crosstalk signal is greater with a peak to peak measurement the receiver is only affected by that portion that deviates from the zero differential level (i.e., the peak level).

Since the crosstalk is a linear function of amplitude it is not required that the actual aggressor signal be the maximum differential amplitude. A scaling technique is used to compensate for equipment that is not capable of launching maximum amplitude signals.

Although crosstalk is generally more intense with shorter STD aggressor signals, both the maximum and minimum STD signals are required to be used. This is to cover the case where physical imperfections may extend over longer distances and therefore could yield a more intense crosstalk with longer STD aggressor signals. Crosstalk does not necessarily scale linearly with STD. Therefore, the specific STD requirements shall be used.

In a SCSI cable aggressor signals on each of the DATA, PARITY, and REQ or ACK pair induces noise on the ACK or REQ pair respectively. Each DATA, PARITY, and REQ or ACK pair shall be separately excited. The induced absolute peak noise (deviation from zero differential) on the ACK or REQ pair measured at a time position not associated with the test fixture is recorded as the crosstalk contribution from that aggressor signal. The results from each aggressor signal are added to yield the total crosstalk.

E.8.2 Sample preparation

This test requires type B samples (see table E.1) as described in section E.2.2.2

E.8.3 Test fixture and measurement equipment

The same basic test fixture is used as for the impedance tests. See E.2.2.3.

The measurement equipment is also the same as for the impedance tests except that a separate receiving head is used for the measuring instrument (SMI1) as shown in E.22.

The length and properties of the 50 Ω leads connecting the aggressor signal and the victim measurement instrument to the test fixture should be the same length.

E.8.4 Calibration procedure

The STD and time reference calibration is done using the same test fixture and nearly the same procedure as for the TDR tests in E.2.2.3 (using a short in place of the DUT for STD and reference time calibration).

Noting the time position of the short establishes a reference time for determining the parts of the test configuration that are causing the crosstalk.

A second calibration fixture configuration is used to verify that the fixture is not causing excessive crosstalk and to verify the time position of the TP1. This second fixture is identical to that described in E.2.2.3 but with 100 Ω resistors added instead of shorts. The second calibration setup is shown in figure E.21.

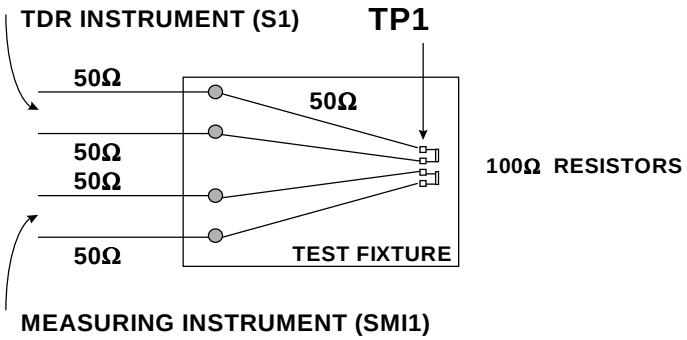


Figure E.21 - Calibration system for NEXT

The amplitude calibration is the same as for the impedance tests in section Differential signal transition duration (STD) calibration. Use the 100 % differential amplitude as defined in figure E.3.

Note the exact settings used for both the minimum and maximum STD conditions as these are reused when doing the actual measurement.

E.8.5 Testing procedure

Using the test setup shown in figure E.22 apply the calibrated aggressor pulse for the minimum STD to the aggressor line, S1, and measure the induced noise on the victim line at measuring instrument (SMI1) as shown in E.22.

Repeat the test exactly except with the polarity of the leads to S1 reversed.

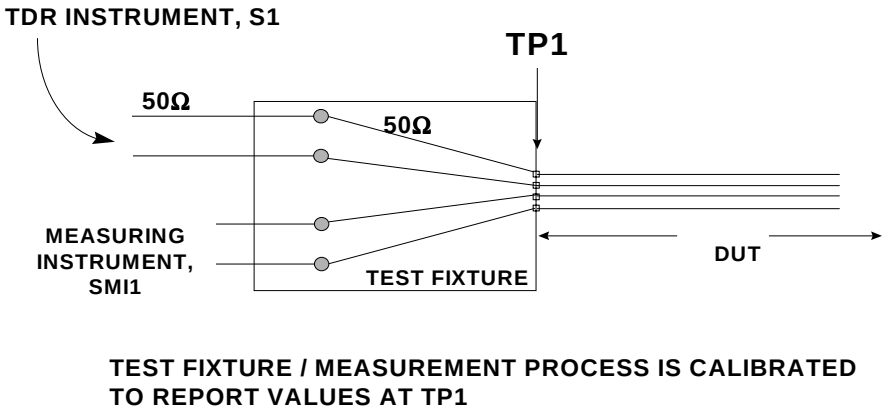


Figure E.22 - Test configuration for NEXT

Note the largest peak (i.e. largest deviation from zero differential) on the victim line at a time position farther from S1 than the time position of the short determined in the calibration. This largest peak from either polarity is the value of the induced signal for that STD. Note that a peak to peak value is not used. Both the absolute value of the induced signal peak and its percentage with respect to the amplitude of the

aggressor signal are recorded.

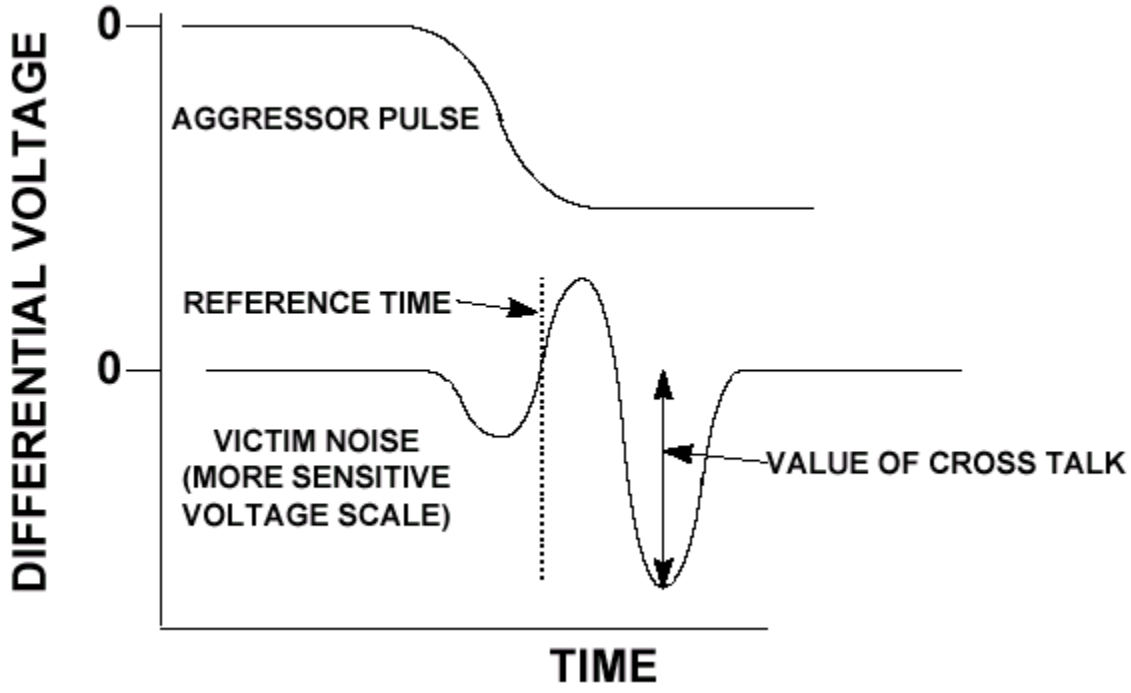


Figure E.23 - Example of crosstalk measurement

Repeat the tests for both polarity of lead attachment to S1 with the maximum STD aggressor signal.

The absolute value crosstalk is scaled to account for the actual amplitude of the aggressor signal. For example if the actual aggressor signal is 500 mV peak and the maximum allowed aggressor signal is 1000 mV then the measured absolute crosstalk result would be multiplied by 2,0.

The percentage result does not need to be scaled.

Crosstalk percent is calculated as follows:

$$\%NEXT = \frac{\Sigma \text{ peak absolute differential induced voltages on REQ or ACK}}{\text{peak to peak differential aggressor voltage}}$$

Software filtering is not allowed for this test - hardware filters are required to produce the rise time required.

Annex F

(normative)

Simple expander requirements

F.1 Introduction

This annex describes extended SCSI configurations that incorporate simple expanders to separate the SCSI domain into electrically isolated parts by using active circuits. The requirements for these active circuits are described in this annex. These active circuits are specified to produce few additional requirements beyond those defined within this standard for initiators and targets. Initiators and targets that are implemented in accordance to requirements in this standard do not produce any behavior that causes extended configurations defined in this annex to fail provided that the extended configuration meets the requirements in this annex.

Initiators shall implement the basic integrity check method for physical layer integrity checking described in annex N.

F.2 Glossary

F.2.0 SCSI domain: A SCSI domain is a logical bus with at least one bus segment, at least one initiator, and at least one target. Domains with multiple bus segments are enabled through the use of bus expanders. Domains consist of the set of SCSI devices that are addressable from an initiator or target.

F.2.0 Simple expander: Devices that couple bus segments together without using SCSI ID's in the device and are "invisible" to initiators and targets.

~~**F.2.0 Bridging expander:** Devices that couple a bus segment to another SCSI segment or another kind of port by using addressable SCSI ports in the device. Bridging expanders have SCSI IDs on all ports, participate in SCSI arbitration and messaging, and are "devices" in the SCSI sense. Bridging expanders are not defined in this standard.~~

F.3 Bus segments in a SCSI domain

SCSI busses are based on the assumption that there is a single electrically conducting path between bus terminators for each signal and that a SCSI domain contains all the devices between these two terminators. This electrical path is assumed to pass signals in both directions without delay other than that caused by the propagation delay of the transmission line associated with the path. It is assumed that there are no intervening active components in the path between the bus terminators.

An alternative to this would be to build SCSI domains that use more complex physical implementations where there may be active electrical components between SCSI devices.

A building block for these alternative implementations is a bus segment that is defined as two bus terminators and the associated single electrically conducting path between these terminators, for each signal, that satisfies the assumptions in the first paragraph of this clause. Multiple bus segments may be functionally connected together by the active circuits described in this annex.

Each bus segment shall:

a) have TERMPWR sources and TERMPWR distribution parameters;
b) use the same transceiver type (LVD or SE) within the segment, however, a SCSI domain may contain segments that use different transceiver types; and
c) follow the same rules that are described in this standards.

Although each bus segment does not extend any SCSI properties the combination of multiple bus segments into a single SCSI domain allows:

a) increased device counts;
b) increased physical length limits;
c) controlling ground voltage shifts;
d) dynamic removal and replacement of portions of SCSI domains; and
e) mixing of SCSI device transceiver types (SE, LVD).

F.4 Simple bus expanders

The following features shall be the required properties of simple bus expanders:

a) No SCSI IDs used;
b) No ARBITRATION phases initiated;
~~c) No messages originating with the simple expander shall be sent by the simple expander, however, messages sent from initiators and targets could be read (e.g., the simple expander could require the negotiated data phase speed or some other variable property of a transaction of an initiator/target connection;~~
d) No messages originating with the simple expander;
e) Retransmitted signals from the simple expander that meet the same requirements as any other SCSI device at the expander boundary. The simple expander boundary may be at a separable connector;
f) Simple expander receivers shall operate error free with the most degraded signal allowed for any SCSI device receiver at the expander boundary. The simple expander boundary that may be at a separable connector;
g) Simple expanders shall not interfere with the REQ/ACK OFFSET count in any initiator or target in the SCSI domain other than that caused by the propagation time through the expander;
h) Simple expanders that are powered on shall retransmit RESET assertions from one bus segment to the other regardless of the state of any other SCSI signals on either side;
i) Simple expanders shall operate with any arbitrary placement of the initiators and targets with respect to the simple expander (e.g., all targets and initiators could be on the same side of the expander or there could be initiators and targets on both sides of the expander);
j) TERMPWR shall not be connected in the expander between the bus segments;
k) DIFFSENS shall not be electrically or logically connected between bus segments; and
l) Transceiver mode changes (e.g., SE to LVD) on one bus segment shall cause the simple expander to issue a hard reset (see 12.3) on the another bus segment.

NOTE 63 - Simple expanders should consume minimal propagation time during arbitration so that the end to end SCSI domain propagation time budget may be used primarily for longer physical length connections (see F.8.3).

Figure F.1 shows a single simple expander between two bus segments.

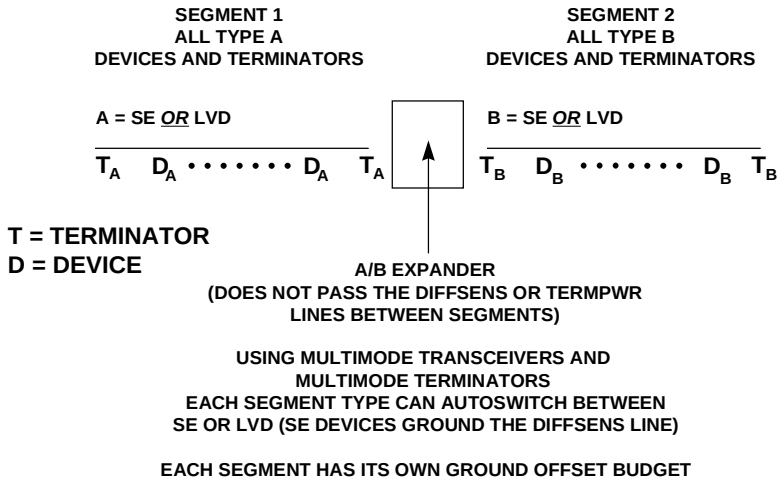


Figure F.1 - A two segment domain using a single simple expander circuit

Figure F.2 shows three ways that simple expanders may be used to connect bus segments.

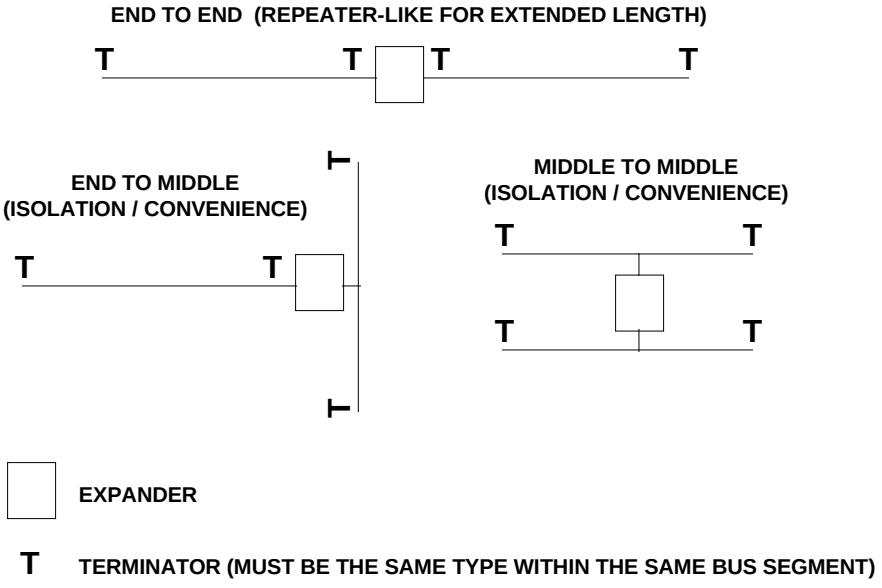


Figure F.2 - Three ways to couple bus segments together with simple expanders

F.5 Homogeneous type

If a simple expander has the same type of bus segment on both sides it is termed a homogeneous expander. The homogeneous expander does not do transceiver type conversions (e.g. SE to LVD).

NOTE 64 - This kind of simple expander may be useful in existing systems where domain length increases may be achieved by inserting a single homogeneous expander in the right place. Such a condition exists, for example, in a SCSI domain where several SE devices are connected to a backplane and subsequently to an initiator by a shielded external cable. By placing a homogeneous expander near the backplane one creates a short, heavily loaded backplane bus segment and a point to point bus

segment to the initiator. Increases in overall SCSI domain physical length may be achieved because the point to point bus segment length limit is longer than the multi-drop bus segment length.

F.6 Heterogeneous types

Simple expanders that have different bus transceiver types on each side are heterogeneous expanders.

F.7 SCSI domain examples using simple expanders

Figure F.3 shows two examples of SCSI domains built using only simple expanders.

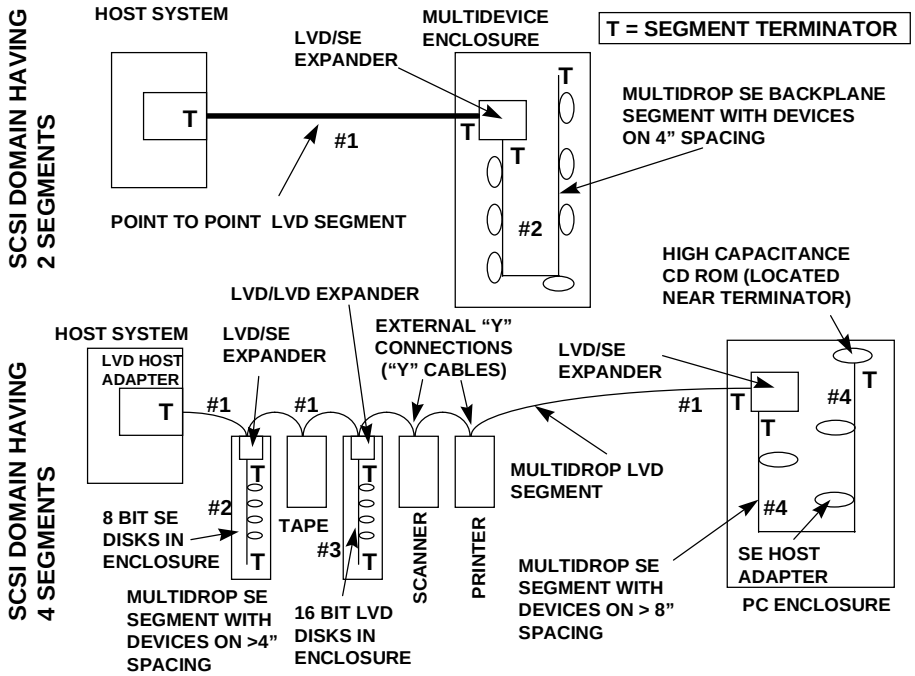


Figure F.3 - Examples of SCSI domains using simple expanders

F.8 Rules for SCSI domains using simple expanders

F.8.1 Rule summary

The rules are summarized in table F.1 followed by detailed discussion for each in the subsequent subclauses.

Valid SCSI domains shall conform with the rules in table F.1.

Table F.1 - SCSI Domain rules

Rule	Description	Clause
1	All bus segments in the domain shall comply with their individual bus segment length limits and other bus segment related requirements.	F.8.2
2	Any bus segment between two other bus segments shall support the highest performance level that can be negotiated between the two other bus segments. (e.g.,two wide LVD Fast-40 bus segments shall not be separated by a bus segment that does not support both wide and fast-40. See figure F.4 for examples.	F.8.3
3	The expander between two bus segments shall support the maximum performance levels supported on each SCSI interface of the simple expander.	F.8.4
4	The maximum propagation time between any two SCSI devices in the SCSI domain shall not exceed 400 ns.	F.8.5
5	The number of addressable SCSI devices in the SCSI domain shall not exceed the addressability of the SCSI devices in the SCSI domain.	F.8.6
6	Loop topologies are not allowed.	F.8.7

F.8.2 Rule 1

Requirements for SCSI domains consisting of a single SE bus segment or a single LVD bus segment are specified in detail in other clauses of this document. Every bus segment in a multi-segment SCSI domain shall conform to the requirements for single bus segment SCSI domains of the same transceiver type.

F.8.3 Rule 2

Rule 2 applies to intermediate bus segments that only exist in SCSI domains of at least three bus segments. The bus segment between the two other bus segments is the intermediate bus segment. The intermediate bus segment shall be wide if both other bus segments are wide. The intermediate bus segment shall support the lowest common transfer rate between the other bus segments.

An example of a rule 2 configuration violation is shown in figure F.4.

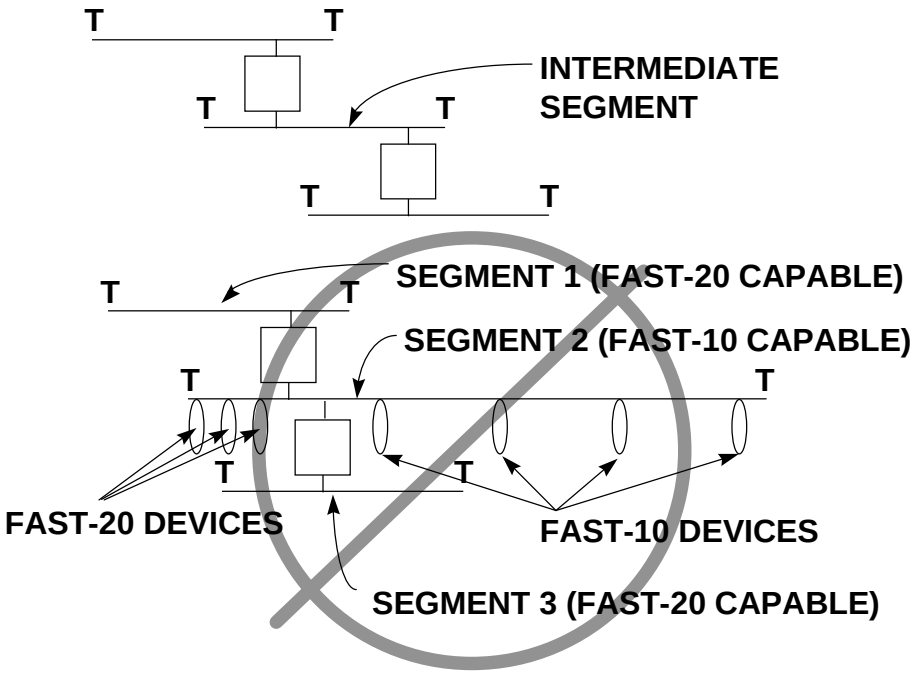


Figure F.4 - Intermediate bus segments and performance ranking

The configuration in figure F.4 is valid only if the transfer rate is limited to fast-10 for any DATA phase transfers between bus segment 1 and bus segment 2, bus segment 2 and bus segment 3, or bus segment 1 and bus segment 3. Even though fast-20 capable SCSI devices in bus segment 2 are located close to the simple expanders and the distance between the simple expanders is small, the bus segment length is defined by the distance between the terminators, not by the distance to the simple expander connection or to the SCSI devices. The intermediate bus segment is not fast-20 capable and may not be used for fast-20 transfer rates between bus segment 1 and bus segment 3. Bus segment 2 shall not be used for fast-20 transfer rates within bus segment 2. Fast-20 transfer rates are allowed between SCSI devices in bus segment 1 or between SCSI devices in bus segment 3.

The intermediate bus segment in this example receives signals at the higher transfer rate on the DATA BUS, DB(P_CRCA), and/or DB(P1) signals, but since the devices in the intermediate bus segment are not participating in the higher transfer rate and are waiting for the next BUS FREE phase, hard reset, or some other phase they are unaffected by the higher transfer rates.

For multimode bus segments, any dynamic transceiver mode change (e.g., LVD to SE) is treated as a fault and the simple expander shall create a hard reset on the segment opposite the one that experienced the transceiver type change. The simple expander shall detect this change by sensing the DIFFSENS line. This scheme ensures that the initiators on the other bus segments are aware of the transceiver mode change and may reassess whether this new transceiver type is consistent with the performance requirements for the bus segments and the overall parameters for the SCSI domain before allowing information transfers to resume. Once a hard reset is detected initiators shall renegotiate with all targets in the SCSI domain.

F.8.4 Rule 3

Homogeneous expanders between two bus segments shall support the maximum performance levels supported on each bus segment of the homogeneous expander. If one bus segment of the homogeneous expander is wide both bus segments of a homogeneous expander shall be wide. Both bus segments of the homogeneous expander shall support the same maximum transfer rate.

F.8.5 Rule 4

F.8.5.1 Effects of wired-or glitches

Wired-or glitches occur when two on more drivers are asserting the same signal line and one subsequently ceases to drive the line. This condition happens frequently during the ARBITRATION phase on the BSY signal and may happen on other wired-or signals. This change in the number of asserted drivers causes a redistribution of current in the segment, with resulting voltage glitches, and may cause false detection of a BUS FREE phase and other errors. The worst case condition is when two SCSI devices near a bus segment terminator are involved. In this case, after the SCSI device stopped asserting the line, it requires a full bus segment length round trip time before the line is again stable . If this condition applies, the round trip time allowed is 400 ns. The one way time is 200 ns.

Waiting the entire domain round trip time may be avoided by ensuring that wired-or glitches do not pass through the simple expander. This standard does not describe how simple expanders implement this capability. If wired-or glitches are not propagated through simple expanders, then the maximum round trip SCSI domain signal propagation time is 800 ns and the one way SCSI domain propagation time is 400 ns.

If a simple expander does not implement the wired-or glitch filter it shall be labeled indicating that it allows propagation of wired-or glitches. This standard assumes that wired-or glitch blocking simple expanders are used and that the maximum SCSI domain round trip time of 800 ns is available.

F.8.5.2 Simple expander propagation delay effects

The simple expander is in series with initiators and/or targets when the path between the initiators and/or targets goes through an expander. In this case the propagation delay through the simple expander shall be counted as part of the 400 ns budget between those SCSI devices.

The delay varies depending on the implementations. ~~Care shall be exercised when considering expanders to understand the capabilities of the expanders being used.~~ When two simple expanders are in series the delay across the pair may be much less than twice the individual delays. This is because the “direction” change that consumes much of the propagation delay during the ARBITRATION phase only applies to one of the simple expanders at a time. The single simple expander delay, T_{ds} and the expander series pair delay, T_{dp} should be specified.

If the simple expander is attached to a bus segment (e.g., as in case of the SCSI device enclosures in the bottom part of figure F.3) it is only in series between the SCSI devices in the enclosure and other SCSI devices in the SCSI domain. The simple expander in the LVD/SE expander in the 8 bit SE enclosure would not be in series between the ~~two~~ host systems LVD host ~~ports~~ adapter and the PC enclosures LVD/SE expander for example.

Editors Note 5 - GOP: The statement above ’The simple expander in the enclosure would not be in series between the two host ports for example.’ was changed to ’The simple expander in the LVD/SE expander in the 8 bit SE enclosure would not be in series between the host systems LVD host adapter and the PC enclosures LVD/SE expander for example.’ is the new wording correct?

The propagation time through the differential transceivers of initiators and targets does not need to be separately accounted for if the wired-or glitches do not propagate through the simple expander. The differential transceiver delay effects are confined to the differential bus segments. Using simple expanders that do not pass the wired-or glitch prevents one bus segment’s delays from being passed on to the next.

F.8.5.3 Sample calculations

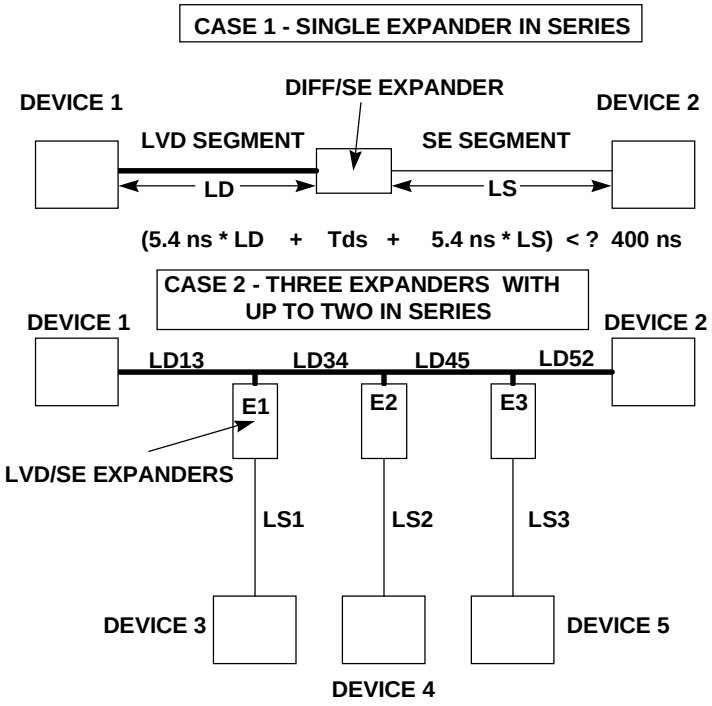


Figure F.5 - Two configurations for SCSI domain delay calculations

Figure F.5 shows two sample SCSI domain configurations. In figure F.5 parameters whose first letter is “L” are physical lengths, “D” refers to differential bus segments and “S” refers to single-ended bus segments. In case 1 the delay calculations are shown in figure F.5. For the more complex case 2 all the possible combinations between any two SCSI devices shall be considered. These calculations are shown in table F.2. The SCSI device pair that has the largest combination of expander propagation time and interconnect propagation time determines if this configuration meets the 400 ns SCSI device to SCSI device maximum propagation time requirement.

Table F.2 - SCSI domain delay calculation

SCSI Device Pair	Path Between SCSI Devices	Simple Expanders Delay (ns)	Interconnect Delay (ns)
1-2	LD13,LD34,LD45,LD52	0	5.4*(LD13+LD34+LD45+LD52)
1-3	LD13,E1,LS1	T_{ds}	5.4*(LD13+LS1)
1-4	LD13,LD34,E2,LS2	T_{ds}	5.4*(LD13+LD34+LS2)
1-5	LD13,LD34,LD45,E3,LS3	T_{ds}	5.4*(LD13+LD34+LD45+LS3)
2-3	LD52,LD45,LD34,E1,LS1	T_{ds}	5.4*(LD52+LD45+LD34+LS1)
2-4	LD52,LD45,E2,LS2	T_{ds}	5.4*(LD52+LD45+LS2)
2-5	LD52,E3,LS3	T_{ds}	5.4*(LD52+LS3)
3-4	LS1,E1,LD34,E2,LS2	T_{dp}	5.4*(LS1+LD34+LS2)
3-5	LS1,E1,LD34,LD45,E3,LS3	T_{dp}	5.4*(LS1+LD34+LD45+LS3)
4-5	LS2,E2,LD45,E3,LS3	T_{dp}	5.4*(LS2+LD45+LS3)

F.8.6 Rule 5

Since simple expanders have no SCSI ID's the maximum number of addressable devices in the SCSI domain is not increased or decreased by the use of simple expanders.

F.8.7 Rule 6

Loop topologies in any form shall not be configured within a SCSI domain. Using simple expanders connected in a loop it is possible to create conditions where both a simple expander and a target or initiator are asserting the same line. Under these conditions the line will not return to the negated state when the initiator or target releases the line since it will continue to be driven by the simple expander. The logic state of the line does not change and a lock up condition exists.

Figures F.6 shows some examples of loops.

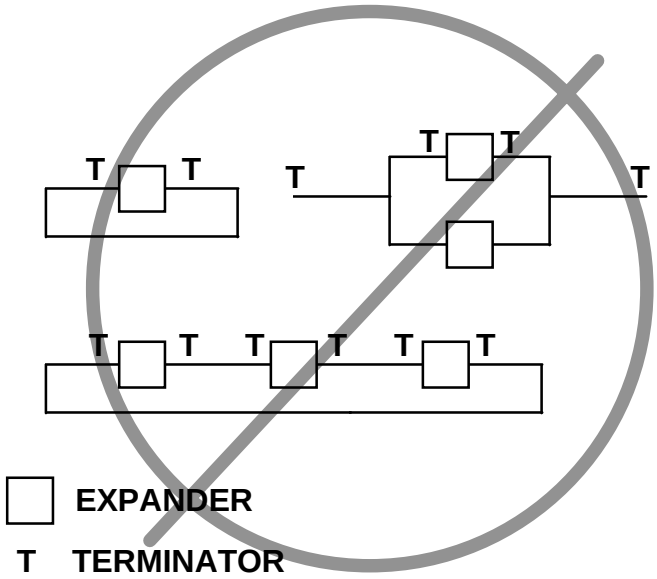


Figure F.6 - Examples of illegal loops

F.9 Special performance considerations for SCSI domains with simple expanders

With simple expanders there is an additional consideration relating to the REQ/ACK OFFSET value. Since the round trip domain propagation time can be as large as 800 ns when using simple expanders, the REQ/ACK OFFSET value negotiated between any two devices should be larger than used for SCSI domains that do not use expanders. If the REQ/ACK OFFSET is not sufficient to accommodate the round trip time between the devices the domain experiences a performance degradation. This minimum REQ/ACK OFFSET value increases with increasing data transfer rates. The minimum REQ/ACK OFFSET value to avoid performance degradation for a variety of conditions are shown in table F.3.

The device REQ/ACK OFFSET counter is set to zero before the DATA phase begins. When a REQ transition occurs the offset counter is incremented. When an ACK transition occurs the counter is decremented. After the DATA phase is completed the offset counter should again be at zero since the number of REQ transitions and the number of ACK transitions should be the same.

When the target sends the first REQ transition there is a minimum of one round trip time before the first ACK transition may be received from the initiator. This round trip time includes the data processing time at the initiator. The target may continue to issue REQ transitions until the offset counter in the target reaches the maximum REQ/ACK OFFSET value that was negotiated.

If the maximum offset value in the target is reached, the target waits until it receives a decrementing ACK transition before issuing another REQ transition. The time spent waiting for a decrementing ACK transition is lost.

The receiving SCSI device is required to accept up to at least the negotiated REQ/ACK OFFSET value of ACK or REQ transitions.

Negotiated REQ/ACK OFFSETs do not affect the operation of simple expanders.

The minimum recommended offset value is given by:

[{2 times one way SCSI domain propagation time} / {ACK(REQ) period}] + processing overhead

Table F.3 gives some representative values from the above equation for round trip SCSI domain propagation times greater than 400 ns assuming the processing overhead to be two ACK(REQ) periods in all cases.

Table F.3 - Minimum REQ/ACK OFFSET for maximum performance

SCSI domain round trip propagation time (ns)	DATA phase speed	ACK(REQ) period (nominal min)	Minimum REQ/ACK OFFSET to avoid performance degradation (assuming two overhead periods in all cases)
500	Fast-10	100	7
600	Fast-10	100	8
700	Fast-10	100	9
800	Fast-10	100	10
500	Fast-20	50	12
600	Fast-20	50	14
700	Fast-20	50	16
800	Fast-20	50	18
500	Fast-40	25	22
600	Fast-40	25	26
700	Fast-40	25	30
800	Fast-40	25	34
500	Fast-80	12.5	42
600	Fast-80	12.5	50
700	Fast-80	12.5	58
800	Fast-80	12.5	66
500	Fast-160	6.25	(82) 84 (note)
600	Fast-160	6.25	(98) 100 (note)
700	Fast-160	6.25	(114) 116 (note)
800	Fast-160	6.25	(130) 132 (note)

Note: Rounded up to the next multiple of 4 - see 16.3.10.1

Annex G

(normative)

Expander Communication Protocol

G.1 Introduction

This annex describes a method of expander communication and topology discovery called Expander Communication Protocol (ECP). This protocol permits application clients to detect expanders that support the protocol. It also permits the application client to pass parameter settings to expanders and permits expanders to report settings and status information. No new hardware features are required of initiators or targets to implement this protocol.

ECP depends on the expander being able to monitor the data transfers associated with WRITE BUFFER and READ BUFFER commands (see SCSI Primary Commands-2 standard) and to alter specific portions of the data transferred as it passes through the expander in either direction. To simplify the expander implementation requirements, ECP is restricted to 8-bit asynchronous transfers.

G.2 Glossary and Definitions

G.2.1 Communicative expander: A simple expander (see Annex F) that has the additional capability to support the requirements of this annex and is capable of transmitting information beyond that received on its ports to specific other entities in the domain. In this annex, unless stated otherwise, the term expander means communicative expander.

G.2.2 Expander function signature: A specific sequence of data bytes that identifies an ECP function in the data phase of a WRITE BUFFER or READ BUFFER command.

G.2.3 Far port: For the current I/O process, an expander port that is not the near port.

G.2.4 First expander: In a series expander set, the expander that couples the bus segment containing the initiator to the next bus segment.

G.2.5 Last expander: In a series expander set, the expander that couples the bus segment containing the target to the next bus segment.

G.2.6 Near port: For the current I/O process, the expander port connected directly to the initiator through a bus segment or connected to the initiator through other expanders and bus segments.

G.2.7 Non-target port: A far port that is not a target port (i.e., all far ports that do not include the target for this I/O process).

G.2.8 n^{th} expander: In a series expander set, the n^{th} expander between the initiator and the target.

G.2.9 Series expander set: The set of one or more expanders that couple the bus segment containing an initiator to the bus segment containing a target.

G.2.10 Target port: For the current I/O process, a far port that is connected directly to the target through a bus segment or connected to the target through other expanders and bus segments.

G.3 Symbols and abbreviations

ECP Expander Communication Protocol

SEDB Short Expander Descriptor Blocks
LEDB Long Expander Descriptor Block

G.4 Enabling ECP

Following a power cycle or bus reset, a communicative expander shall function as a simple expander for each initiator until the initiator enables ECP as follows:

1) negotiate asynchronous transfer mode and a transfer width of 8 bits to some target; and
2) issue a WRITE BUFFER command to that same target with the MODE field set to echo buffer plus enable ECP (see SCSI Primary Commands-2 standard).

The initiator may disable ECP by:

1) negotiating to asynchronous transfer mode with a transfer width of 8 bits to any target; and
2) issuing a WRITE BUFFER command to that same target with the MODE field bit set to disable ECP (see SCSI Primary Commands-2 standard).

This enabling and disabling of ECP is done on an initiator basis (i.e., each initiator issues a single WRITE BUFFER command to enable or disable ECP for that initiator for all communicative expanders on the bus). The enabling or disabling of ECP occurs regardless of the device server's response to the WRITE BUFFER command.

While the initiator has ECP enabled, it is responsible for not issuing WRITE BUFFER commands with the EXPANDER FUNCTION SIGNATURE in the first 7 bytes of the data buffer unless it is an expander function header (see G.5).

G.5 Communicative expander function structures

Communicative expander functions consist of outbound and inbound functions. The outbound functions are contained in the data of a WRITE BUFFER command with the MODE field set to write data mode, echo buffer mode or echo buffer plus enable ECP mode. The inbound functions return information in the data of a READ BUFFER command with the mode field set to data mode, echo buffer mode or echo buffer plus enable ECP mode.

The initiator shall not enable disconnects for these WRITE BUFFER and READ BUFFER commands (i.e., the DISCPRIV bit in the IDENTIFY message (see 16.3.3) is set to 0).

The outbound and inbound functions are further divided into multiple and single functions.

For multiple functions, the data is transferred in a 172-byte data structure consisting of a 16-byte expander function header followed by ten 16-byte SEDB. For single functions, the data is transferred in a data structure whose length depends on the selected expander function code. This data structure consists of a 16-byte expander function header followed by one LEDB. In either case, the first 16 bytes of the data structure contain an expander function header as shown in table G.1.

Table G.1 - Expander function header

Bit Byte	7	6	5	4	3	2	1	0
0	MSB	EXPANDER FUNCTION SIGNATURE (B73384B8508F27h)						
6								LSB
7	INITIATOR SCSI ADDRESS							
8	EXPANDER FUNCTION CODE							
9	MSB	Function specific						
15								LSB

The EXPANDER FUNCTION SIGNATURE contains a code of B73384B8508F27h that signifies this WRITE BUFFER or READ BUFFER data is an expander function.

The application client shall set the INITIATOR SCSI ADDRESS field to the SCSI address of the initiator through which the command is to be sent (i.e., the I of the I_T nexus). The expander shall check that this field matches the SCSI address for the initiator of the current I/O process.

If both the expander function signature is correct and the INITIATOR SCSI ADDRESS field matches the initiator's SCSI address, then this WRITE BUFFER or READ BUFFER data is an expander function that shall be processed by the expander. Otherwise, the WRITE BUFFER or READ BUFFER command shall be ignored by the communicative expanders. In all cases the communicative expanders shall repeat the WRITE BUFFER or READ BUFFER data.

The EXPANDER FUNCTION CODES are described in table G.2.

The function specific bytes are documented for single functions in G.6.2 and G.6.4. The function specific bytes are not presently used for multiple functions.

Table G.2 - Expander functions

Type	Code	Expander function	Clause
Outbound multiple function	00h	ASSIGN ADDRESS	G.6.1.2
	01h	MARGIN CONTROL	G.6.1.3
	02h - 2Fh	Reserved	
	30h - 3Fh	Vendor specific	
Outbound single function	40h	CONTROL	G.6.2.2
	41h - 6Fh	Reserved	
	70h - 7Fh	Vendor specific	
Inbound multiple function	80h	Reserved	
	81h	MARGIN REPORT	G.6.3.2
	82h	REPORT CAPABILITIES	G.6.3.3
	83h - AFh	Reserved	
	B0h - BFh	Vendor specific	
Inbound single function	C0h	EXPANDER INQUIRY	G.6.5
	C1h - EFh	Reserved	
	F0h - FFh	Vendor specific	

For outbound and inbound multiple functions, the expander function header is followed by ten short expander descriptor blocks (see table G.3).

For outbound single functions (see G.6.2) and inbound single functions (see G.6.4), the expander function header is followed by one long expander descriptor block.

Table G.3 - SEDB

Bit Byte	7	6	5	4	3	2	1	0
0	USED	RESERVED				D_CLASS		
1	Function specific fields							
15								

The USED bit is defined in G.6.1.1, G.6.2.1, G.6.3.1, and G.6.4.1.

The D_CLASS field identifies the device class that set the USED bit to one as defined in table G.4:

Table G.4 - Device class

Code	Device class
000b	Reserved
001b	Communicative expander device
010b	SCSI initiator device
011b - 111b	Reserved

The remaining fields in the short expander descriptor block are specific to the expander function and are defined in G.6.

G.6 Expander functions

G.6.1 Outbound multiple functions

G.6.1.1 Outbound multiple function data transfer rules

Outbound multiple functions shall be performed during a WRITE BUFFER command with the MODE field set to one of the values specified in table G.2. The initiator shall transfer an expander function header followed by ten SEDBs. The application client may use an SEDB to indicate the initiator's parameters or settings. If the application client uses an SEDB, it shall use the first SEDB and shall set the USED bit in the first SEDB to 1. The application client shall set the USED bit to 0 in all unused SEDBs.

Each expander in the SCSI domain shall repeat the entire data structure without alteration to its target port, if any, except the expander shall alter the first SEDB encountered with a USED bit of 0. In this SEDB, the expander shall set the USED bit to 1, shall set the D_CLASS field as described in table G.4, and shall output a zero in the reserved field of the first byte. The remaining 15 bytes of this SEDB shall be repeated without alteration. The expander shall interpret the other fields of this altered SEDB as defined in the ASSIGN ADDRESS (see G.6.1.2), and MARGIN CONTROL (see G.6.1.3) expander function codes.

Each expander in the domain shall either repeat the data received on the near port or shall repeat the data output to the target port on all far ports that are not part of the I_T nexus.

An expander that receives a reserved or unimplemented vendor specific multiple expander function code shall follow all of the rules in this sub-clause, but shall ignore the contents of the altered SEDB.

G.6.1.2 ASSIGN ADDRESS

The ASSIGN ADDRESS expander function is used to assign an expander address to one or more expanders for this initiator. The expander address is specific to the assigning initiator. The expander address assigned by one initiator has no affect on the expander addresses assigned by other initiators. The SEDB for this expander function is shown in table G.5.

Table G.5 - ASSIGN ADDRESS SEDB

Bit Byte	7	6	5	4	3	2	1	0
0	USED	RESERVED				D_CLASS		
1	ASSIGN	EXPANDER ADDRESS						
2	RESERVED							
15								

An ASSIGN bit of 1 indicates that the expander shall respond to the expander address specified in the EXPANDER ADDRESS field for single functions from this initiator. The address assignment shall remain in effect until changed by another ASSIGN ADDRESS function from this initiator or until the next reset condition or power cycle. An ASSIGN bit of 0 indicates that the expander shall not change its expander address assignment for this initiator.

Assigning the expander address 0000000b to an expander shall indicate that it has no expander address assigned for this initiator. The application client shall not use expander address 0000000b in single functions.

G.6.1.3 MARGIN CONTROL

The MARGIN CONTROL expander function sets parameter settings in the initiator or expander for usage between the I_T nexus on subsequent synchronous transfers and paced transfers. These parameter settings shall remain in effect until changed by another MARGIN CONTROL expander function or by a reset condition.

The MARGIN CONTROL SEDB is shown in table G.6.

Table G.6 - MARGIN CONTROL SEDB

Bit Byte	7	6	5	4	3	2	1	0
0	USED	RESERVED				D_CLASS		
1	DRIVER STRENGTH (near port)				RESERVED			
2	SIGNAL GROUND BIAS (near port)				DRIVER PRECOMPENSATION (near port)			
3	SLEW RATE (near port)				RESERVED			
4	RESERVED				RESERVED			
5	RESERVED				RESERVED			
6	RESERVED				RESERVED			
7	VENDOR SPECIFIC (near port)				VENDOR SPECIFIC (near port)			
8	RESERVED							
9	DRIVER STRENGTH (far port)				RESERVED			
10	SIGNAL GROUND BIAS (far port)				DRIVER PRECOMPENSATION (far port)			
11	SLEW RATE (far port)				RESERVED			
12	RESERVED				RESERVED			
13	RESERVED				RESERVED			
14	RESERVED				RESERVED			
15	VENDOR SPECIFIC (far port)				VENDOR SPECIFIC (far port)			

Two duplicate sets of margin control fields (i.e., DRIVER STRENGTH, SIGNAL GROUND BIAS, DRIVER PRECOMPENSATION, and SLEW RATE) are provided, one set for the near port and another set for the far port. If the first SEDB is used for the initiator settings, only the far port fields are used and the near port fields are reserved.

The margin control fields shall be implemented as two's complement values with 0000b being the nominal value. The maximum supported setting for each field shall be 0111b and the minimum supported setting for each field shall be 1111b. Up to 16 distinct values are available for each field. Devices that support fewer than 16 distinct values for a field should round non-supported settings to a supported value.

For SIGNAL GROUND BIAS fields, values 0000b through 0111b shall enable the bias cancellation circuit and values 1000b through 1111b shall disable the bias cancellation circuit, if disabling of this circuit is supported.

G.6.2 Outbound single functions

G.6.2.1 Outbound single function data transfer rules

Outbound single functions shall be performed during a WRITE BUFFER command with the MODE field set to one of the single functions values specified in table G.2. The initiator shall transfer an expander function

header followed by a LEDB.

Each expander in the SCSI domain shall repeat the entire data structure without alteration to its target port, if any, except if the USED bit is 0 and the contents of the EXPANDER ADDRESS field in the first byte of the LEDB matches the expanders currently assigned expander address for this initiator (see G.6.1.2). In this LEDB the expander shall change the USED bit to 1 and shall output its currently assigned expander address for this initiator in the EXPANDER ADDRESS field. The expander shall interpret the other fields of this altered LEDB as defined in the CONTROL (see G.6.2.2) expander function code.

Each expander in the domain shall either repeat the data received on the near port or shall repeat the data output to the target port on all far ports that are not part of the I_T nexus.

An expander that receives a reserved or unimplemented vendor specific single expander function code shall follow all of the rules in this sub-clause, but shall ignore the contents of the altered LEDB.

G.6.2.2 CONTROL

The CONTROL function is used to set or clear parameters on the addressed expander. The function specific bytes in the expander function header are reserved for the CONTROL function. The SEDB for the CONTROL function is shown in table G.7.

Table G.7 - CONTROL data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	EXPANDER ADDRESS						
1	TARGET_ADRS							
2	RESERVED					FAR_CTL		
3	RESERVED							
15								

The TARGET_ADRS field shall be set to the SCSI address of a target connected to one of the expander's far ports. This identifies the far port to be controlled on expanders that have multiple far ports. If the specified TARGET_ADRS does not match a known target SCSI address, then the expander shall perform no far port control action on any port.

The FAR_CTL field is defined as shown in table G.8.

Table G.8 - FAR_CTL field values

Code	Far port control action	Description
000b	No operation	Shall have no effect on the specified far port
001b	Disable far port	Shall cause the expander to stop repeating signals to the specified far port and shall cause the expander to ignore signals from the specified far port upon the next BUS FREE phase.
010b	Enable far port	Shall cause the expander to resume repeating signals to the specified far port and shall cause the expander to resume responding to signals from the specified far port upon the next BUS FREE phase.
100b	Reset far port	Shall cause the expander to create a hard reset condition on the specified far port upon the next BUS FREE phase on the near port (i.e., the expander creates a pulse on the RST signal). The expander shall not propagate this hard reset condition to any other of its ports.
all other codes	Reserved	

G.6.3 Inbound multiple functions

G.6.3.1 Inbound multiple function data transfer rules

The application client shall set the USED bit in each of the SEDBs to 0. The data structure containing an inbound multiple function is then placed in the target's buffer using a WRITE BUFFER command with the MODE field set to one of the inbound multiple function values specified in table G.2. Expanders shall not alter the data structure during the WRITE BUFFER command if the EXPANDER FUNCTION CODE is 80h to FFh. The inbound multiple function is then performed during a subsequent READ BUFFER command with the MODE field set to one of the inbound multiple function values specified in table G.2.

During the data transfer phase of the READ BUFFER command, each expander with a target port shall repeat the entire data structure without alteration to its near port, except the expander shall alter the first SEDB encountered with a USED bit of 0. In this SEDB, the expander shall change the USED bit to 1, shall set the D_CLASS field as described in table G.4, and shall output zero in the reserved field of the first byte. The remaining 15 bytes of this SEDB shall be output as described in the MARGIN REPORT (see G.6.3.2) and REPORT CAPABILITES (see G.6.3.3) expander function codes.

Each expander in the domain shall either repeat the data received on the target port or shall repeat the data output to the near port on all far ports that are not part of the I_T nexus.

An expander that receives a reserved or unimplemented vendor specific multiple EXPANDER FUNCTION CODE shall follow all of the rules in this sub-clause, but shall output 00h in bytes 1 through 15 of the altered SEDB.

G.6.3.2 MARGIN REPORT

The MARGIN REPORT expander function is used to report the current margin settings for the initiator or expander. The MARGIN REPORT SEDB is shown in table G.9.

Table G.9 - MARGIN REPORT SEDB

Bit Byte	7	6	5	4	3	2	1	0
0	USED	RESERVED				D_CLASS		
1	DRIVER STRENGTH (near port)				RESERVED			
2	SIGNAL GROUND BIAS (near port)				DRIVER PRECOMPENSATION (near port)			
3	SLEW RATE (near port)				RESERVED			
4	RESERVED				RESERVED			
5	RESERVED				RESERVED			
6	RESERVED				RESERVED			
7	VENDOR SPECIFIC (near port)				VENDOR SPECIFIC (near port)			
8	RESERVED							
9	DRIVER STRENGTH (far port)				RESERVED			
10	SIGNAL GROUND BIAS (far port)				DRIVER PRECOMPENSATION (far port)			
11	SLEW RATE (far port)				RESERVED			
12	RESERVED				RESERVED			
13	RESERVED				RESERVED			
14	RESERVED				RESERVED			
15	VENDOR SPECIFIC (far port)				VENDOR SPECIFIC (far port)			

Two duplicate sets of margin report fields (i.e., DRIVER STRENGTH, SIGNAL GROUND BIAS, DRIVER PRECOMPENSATION, and SLEW RATE) are provided, one set for the near port and another set for the far port. If the last SEDB is used to communicate initiator settings to the application client, only the far port fields are used while the near port fields are reserved.

The margin report fields shall return the current settings for the I_T nexus. Fields that are not implemented shall be reported as 0000b. Otherwise, the current setting for the field, possibly rounded as described in G.6.1.3, shall be returned.

G.6.3.3 REPORT CAPABILITES

The REPORT CAPABILITES function may be used to determine domain topology and report expander characteristics. The REPORT CAPABILITES SEDB is shown in table G.10.

Table G.10 - REPORT CAPABILITES SEDB

Bit Byte	7	6	5	4	3	2	1	0
0	USED	RESERVED				D_CLASS		
1	MSB	FAR SCSI ID LIST						
2								LSB
3	MINIMUM TRANSFER PERIOD FACTOR							
4	RESERVED							
5	MAXIMUM REQ/ACK OFFSET							
6	MAXIMUM TRANSFER WIDTH EXPONENT							
7	PROTOCOL OPTION BITS SUPPORTED							
8	PORTS				RESERVED		TARG_MODE	
9	RESERVED							
15								

The FAR SCSI ID LIST field contains the inclusive-or of all SCSI IDs known to be located on the target port of the expander (e.g., if SCSI devices with IDs 0, 1, and 12 were previously accessed on the target port, the expander sets this field to 1003h).

The MINIMUM TRANSFER PERIOD FACTOR field shall be set to the smallest value of the TRANSFER PERIOD FACTOR (see 16.3.10.1) supported by the expander.

The MAXIMUM REQ/ACK OFFSET field shall be set to the largest value of the REQ/ACK OFFSET (see 16.3.10.1) supported by the expander.

The MAXIMUM TRANSFER WIDTH EXPONENT field shall be set to the largest value of the TRANSFER WIDTH EXPONENT (see 16.3.10.1) supported by the expander.

The PROTOCOL OPTIONS BITS SUPPORTED field shall set the corresponding bit to one for each supported protocol option bit in byte 7 of the PPR message (see 16.3.10.1).

The PORTS field shall contain the number of ports on the expander not including the near port. A value of 0 in this field indicates that the expander is not capable of reporting this information.

The TARG_MODE field specifies the current bus mode for the target port as defined in table G.11.

Table G.11 - TARG_MODE **field values (far port bus mode)**

Code	Target port bus mode (far port)
00b	Unknown (expander not capable of reporting bus mode)
01b	Single ended
10b	Low Voltage Differential
11b	High Voltage Differential

G.6.4 Inbound single functions

G.6.4.1 Inbound single function data transfer rules

The application client shall set the USED bit in the LEDB to 0. The data structure containing an inbound single function is then placed in the target's buffer using a WRITE BUFFER command with the MODE field set to one of the inbound single function values specified in table G.2. Expanders shall not alter the data structure during the WRITE BUFFER command if the EXPANDER FUNCTION CODE is 80h to FFh. The inbound single function is then performed during a subsequent READ BUFFER command with the MODE field set to one of the inbound single function values specified in table G.2.

During the data transfer phase of the READ BUFFER command, each expander with a target port shall repeat the entire data structure without alteration to its near port, except if the USED bit is 0 and the EXPANDER ADDRESS field in the first byte of the LEDB matches its currently assigned expander address (see G.6.1.2). In this LEDB, the expander shall change the USED bit to 1 and shall output its currently assigned expander address in the EXPANDER ADDRESS field. The expander shall output the remaining bytes of this LEDB as described for the EXPANDER INQUIRY expander function code.

Each expander in the domain shall either repeat the data received on the target port or shall repeat the data output to the near port on all far ports that are not part of the I_T nexus.

An expander that receives a reserved or unimplemented vendor specific single EXPANDER FUNCTION CODE shall follow all of the rules in this sub-clause, but shall output 00h in remaining bytes of the altered LEDB.

G.6.5 EXPANDER INQUIRY

The EXPANDER INQUIRY function is used to report information about the specified expander in a manner similar to the SCSI INQUIRY command (see SCSI Primary Commands-2 standard). The expander function header for this function shall include function specific fields as described in table G.12.

Table G.12 - EXPANDER INQUIRY expander function header

Bit Byte	7	6	5	4	3	2	1	0
0	MSB	EXPANDER FUNCTION SIGNATURE (B73384B8508F27h)						
6								LSB
7	INITIATOR SCSI ADDRESS							
8	EXPANDER INQUIRY FUNCTION CODE (C0h)							
9	RESERVED							
10	PAGE CODE							
11	RESERVED							
12	ALLOCATION LENGTH							
13	RESERVED							
15								

An enable vital product data (EVPD) bit of one specifies that the expander shall return the optional vital product data specified by the PAGE CODE field (see SCSI Primary Commands-2 standard). If the expander does not support the optional vital product data, then it shall return zero in all the bytes specified by the allocation length. A EVPD bit of zero specifies the expander shall return EXPANDER INQUIRY data as described in table G.13.

Table G.13 - EXPANDER INQUIRY data

Bit Byte	7	6	5	4	3	2	1	0
0	USED	ADDRESS						
1	RESERVED							
3								
4	ADDITIONAL LENGTH (33h)							
5	RESERVED							
7								
8	MSB	VENDOR IDENTIFICATION						
15								LSB
16	MSB	PRODUCT IDENTIFICATION						
31								LSB
32	MSB	PRODUCT REVISION LEVEL						
35								LSB
36	VENDOR SPECIFIC							
55								

The VENDOR IDENTIFICATION, PRODUCT IDENTIFICATION, and PRODUCT REVISION LEVEL fields shall return the STANDARD INQUIRY data format (see SCSI Primary Commands-2 standard).

G.7 Data Transfer Requirements

The communicative expander functions shall only be performed when the data transfer agreement is 8-bit asynchronous. For any other data transfer agreement, the communicative expander shall operate as a simple expander.

When altering data, communicative expanders shall construct correct parity for the altered data on the outgoing port.

Annex H

(informative)

Interconnecting buses of different widths

A problem may occur when mixing SPI devices with SCSI-2 devices. The TERMPWR requirements (see table 27) of SPI have been increased to support a 16-bit DATA BUS. SCSI-2 devices may not supply sufficient TERMPWR. An additional source of TERMPWR (e.g., a SPI device) may be required.

When busses of dissimilar width are adapted to one another as shown in figure H.1 for SE and figure H.2 for LVD/MSE, the DATA BUS signals from the wider of the two busses that end at the adapter should be terminated at the adapter. The connectors are designed such that A and P shielded connectors do not intermate directly.

Two of the RESERVED lines (A cable contact numbers 23 and 24) and the OPEN line (A cable contact number 25) on the A cable are TERMPWR lines on the P cable (P cable contact numbers 33, 34, and 35).

8-bit SCSI devices that are connected to the SE P cable should leave the following 9 signals open: DB(8-15), DB(P1).

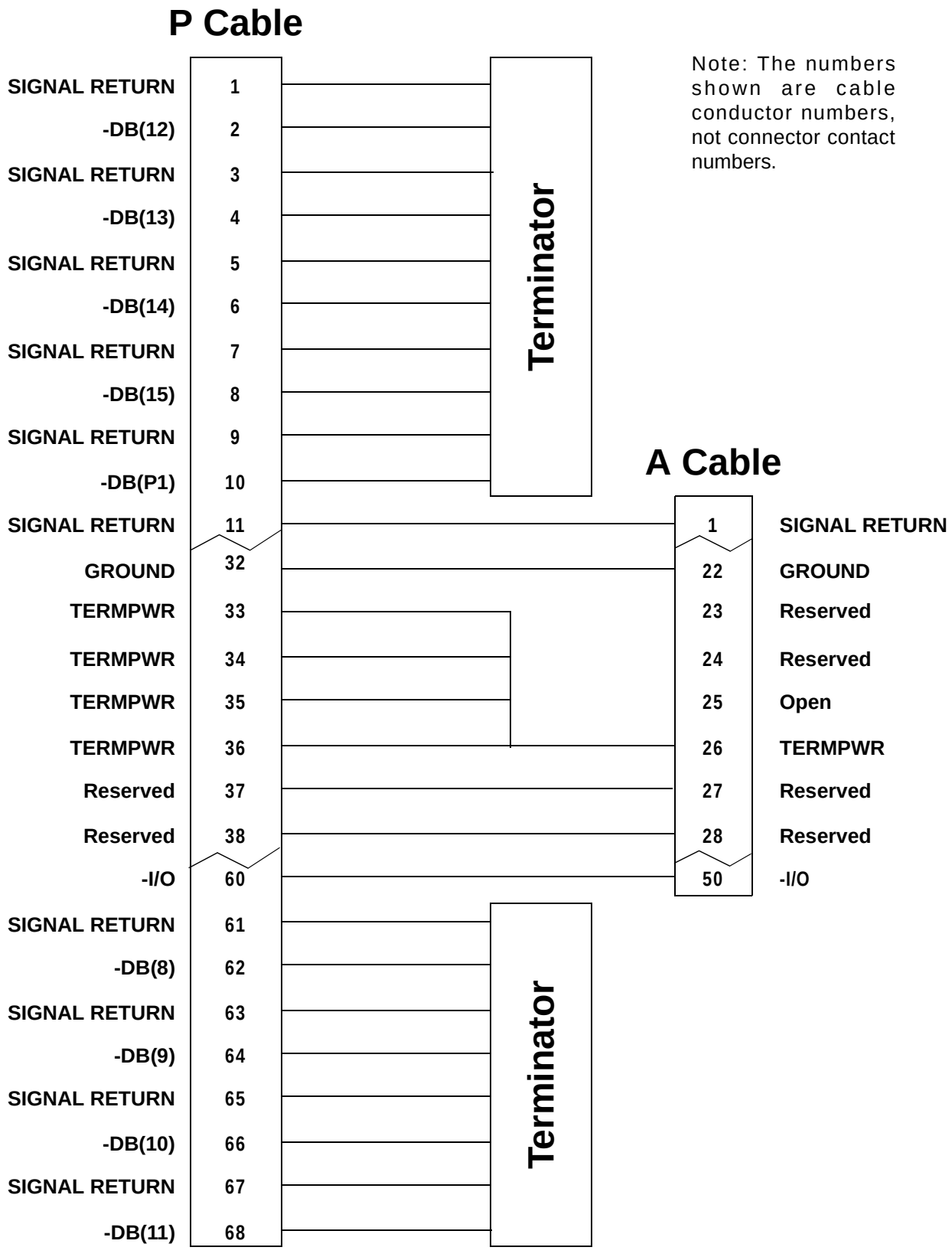


Figure H.1 - Interconnecting SE A and P cables

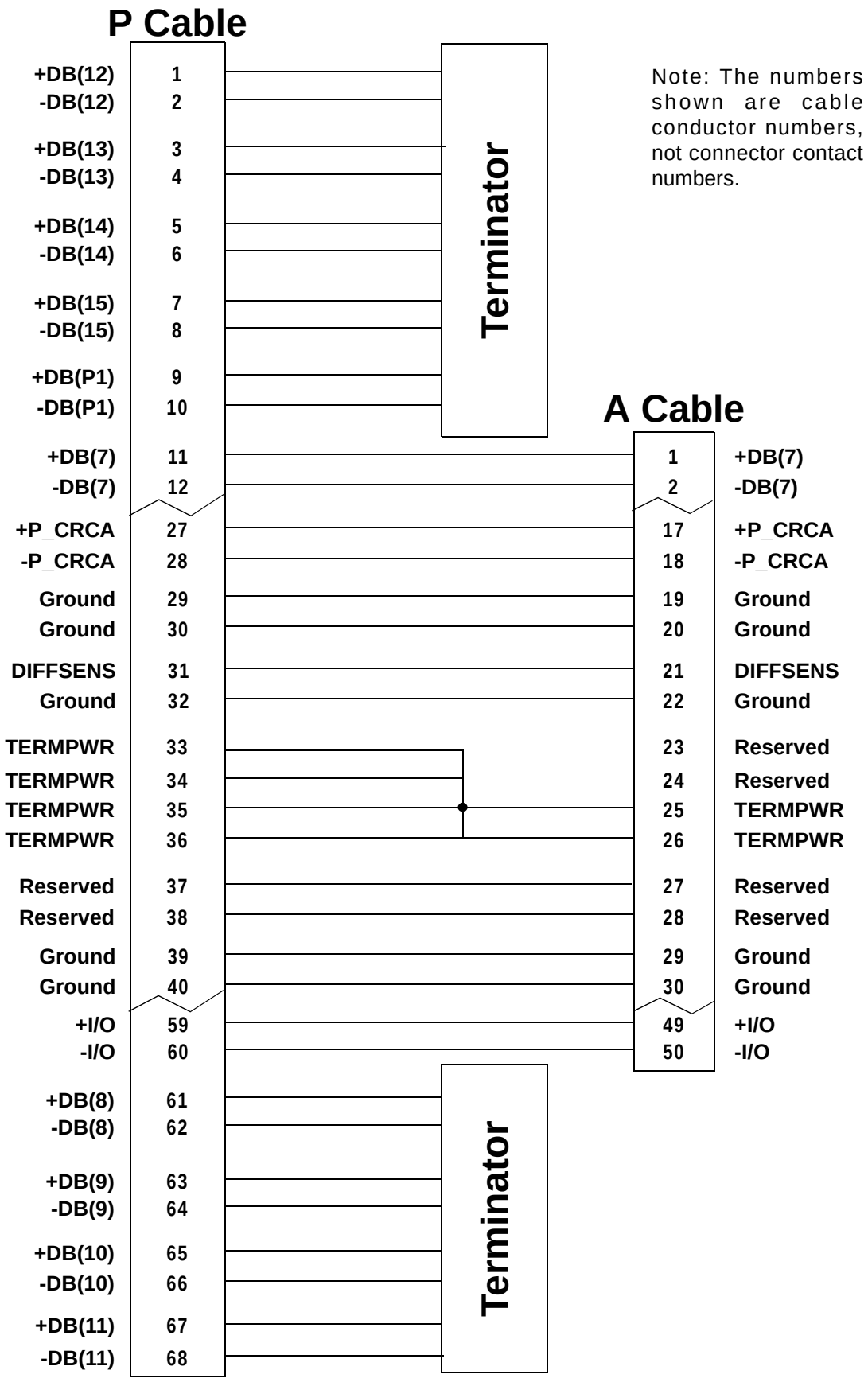


Figure H.2 - Interconnecting LVD/MSE A and P cables

Annex I

(informative)

Transmission line considerations for SE fast-20 data transfer rates

The SCSI bus is a distributed parameter circuit whose electrical characteristics and responses are primarily defined by the distributed inductance and capacitance along the physical media. The media is defined here as the interconnecting cable(s) or conducting paths, connectors, terminators, and SCSI devices added along the bus. The following analysis derives a guideline for the amount of capacitance (and its spacing) that may be added to the SE SCSI buses running up to fast-20 data transfer rates.

To a good approximation, the characteristic transmission line impedance seen into any cut point in the unloaded SCSI bus is defined by $Z = \sqrt{\frac{L}{C}}$, where L is the inductance per unit length and C is the capacitance per unit length. As capacitance is added to the bus, in the form of SCSI devices and their interconnection, the bus impedance is lowered and is expressed by $Z' = \sqrt{\frac{L}{(C + C')}}$, where C' is the added capacitance per unit length. When capacitance is added to the bus by SCSI devices, an impedance mismatch occurs. When a signal wave arrives at this mismatch in impedance, an attenuation (or amplification) of the signal occurs. The magnitude of the attenuation depends upon the ratio of the mismatched impedance or $A = \frac{Z'}{Z}$, where Z' is the load impedance and Z is the source impedance.

Substituting the equations for Z' and Z and reducing,

$$1)\ A = \frac{Z'}{Z} = \frac{\sqrt{\frac{L}{(C + C')}}}{\sqrt{\frac{L}{C}}} = \sqrt{\frac{1}{\left(1 + \frac{C'}{C}\right)}}$$

We now have a relationship for the attenuation of the signal voltage at an impedance mismatch due to load capacitance distributed on the SCSI bus. Next, a rule for the ratio of Z' to Z is derived.

With fast transfer rates and electrically long[1] media, it becomes essential to achieve a valid input voltage level on the first signal transition from an output driver anywhere on the bus. This is called incident-wave switching. If incident-wave conditions are not achieved, reflected-wave switching is used. Reflected-wave switching depends upon reflected energy occurring some time after the first transition arrives to achieve a valid logic voltage level.

In an environment with fast-20 data transfer rates, the valid low-level input voltage threshold has been raised and the high-level input voltage threshold has been lowered to allow incident-wave switching with some inevitable impedance mismatching and signal attenuation along the media.

The signal voltage at an impedance mismatch is $V_{L1} = V_{L0} + V_{J1} + V_{R1}$, where V_{L0} is the initial voltage, V_{J1} is the input signal voltage, and V_{R1} is the reflected voltage. The voltage reflected back from the

1. Electrically long is defined here as $\tau > \frac{t_{10-90\%}}{3}$, where τ is the one-way time delay across the bus and $t_{10-90\%}$ is the 10 % to 90 % transition time of the fastest driver output signal.

mismatch is $V_{R1} = \rho_L \times V_{J1}$ where, $\rho_L = \frac{Z' - Z}{Z' + Z}$ and is the coefficient of reflection commonly used in transmission line analysis. The voltage equation is now written as $V_{L1} = V_{L0} + V_{J1} + (\rho_L \times V_{J1})$.

When a SCSI signal is asserted, the V_{L0} may be at a maximum of 3,7 V and go to 0 V (for a perfect driver) giving a V_{J1} of -3,7 V and the signal voltage should go below the minimum receiver input voltage threshold of 1 V. In equation form,

$$1 > (3,7) + (-3,7) + (\rho_L \times (-3,7))$$

$$\rho_L > \frac{1 - 3,7 + 3,7}{-3,7} = -0,27 \quad .$$

The negative value means that no more than 27 % of the input signal voltage is reflected back towards the source or the minimum assertion level is not achieved by the incident wave[1].

Now, to relate this to Z'/Z and solving equation 1) for C'/C ,

2)
$$\rho_L = \frac{Z' - Z}{Z' + Z} > -0,27$$

$$Z' - Z > -0,27 \times (Z' + Z)$$

$$Z' \times (1 + 0,27) > Z \times (1 - 0,27)$$

$$\frac{Z'}{Z} > (0,73/1,27) = 0,57$$

and

$$0,57 < \sqrt{\frac{1}{\left(1 + \frac{C'}{C}\right)}}$$

$$\frac{1}{0,57^2} - 1 > \frac{C'}{C}$$

$$\frac{C'}{C} < 2,08$$

Therefore capacitance should not be added at more than twice the bus-distributed capacitance for incident-wave switching. For example, a cabled bus with L = 295 nH/m (90 nH/ft) and C = 41 pF/m (12,5 pF/ft) and Z = 85 Ohms, the guideline becomes to add no more than 85 pF/m (26 pF/ft) anywhere along the bus. This guideline is met by 25 pF loads spaced 0,3 m (1 ft) from each other, 50 pF spaced 0,6 m (2 ft) apart, or 12,5 pF spaced 0,15 m (0,5 ft) apart. This relationship is shown graphically in figure I.1.

1. A similar analysis may be used for the negation case of 0 V to 2,8 V ([48 mA + 22 mA] x 40 Ω) and an input voltage threshold of 1,9 V for a minimum reflection coefficient of -0,32. This leaves assertion as the most restrictive case.

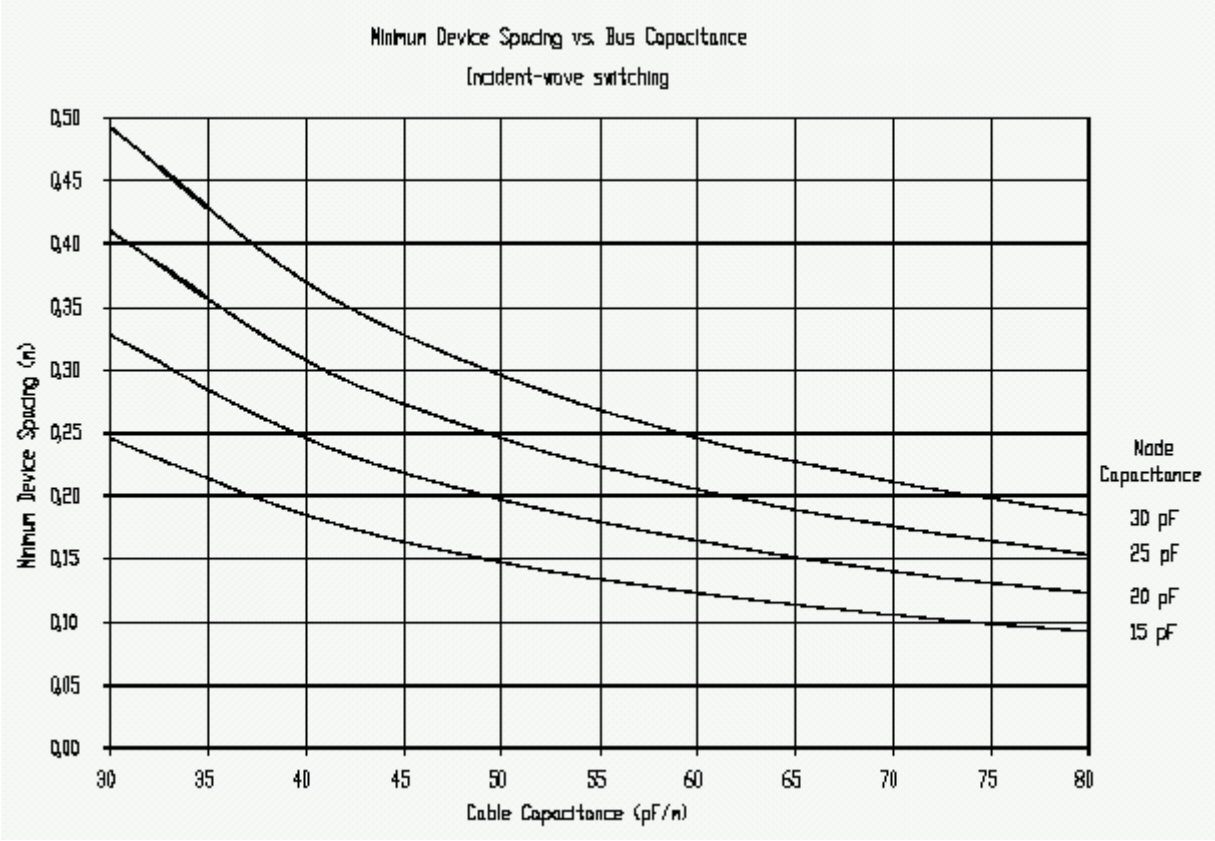


Figure I.1 - Minimum SCSI device spacing versus bus and SCSI device capacitance

Annex J

(informative)

Measuring SE pin capacitance

The objective of this procedure is to determine the lumped capacitance imposed on each signal conductor of the bus proper by a SCSI device connected thereto. The model for this procedure assumes the bus in ribbon cable form passing through an insulation-displacement SCSI connector, the mating part that is mounted on a SCSI device controller printed-wire board. The bus connector is removed from the SCSI device, along with every source of power.

One or more device connector circuit-common pins are connected together to form an effective circuit-common node. An R-F admittance bridge (or equivalent), operation at 1,0 MHz, is connected successively to each signal pin in the device connector, with reference to the circuit-common node.

The signal applied during measurement is biased to 0,5 V D.C. and is 0,4 V peak-to-peak in amplitude.

The characteristics are determined in terms of a parallel combination of a conductance and a capacitive susceptance. The corresponding capacitance thus determined is the maximum signal capacitance referred to in table 19.

NOTE 65 - SCSI signals contain a wide range of frequency components, so that it is not practical to "tune" a bus conductor by loading it with shunt inductance. Consequently, this procedure is performed without any inductive element connected.

Annex K

(informative)

SCSI ICONS

These icons are provided as symbols to identify a SCSI port and to indicate whether the port is using:

a) SE transceivers (figure K.1),
b) LVD transceivers (figure K.2), or
c) SE/LVD multimode transceivers (figure K.3).

The icons illustrated in figure K.1, figure K.2, and figure K.3 may be enlarged or reduced as needed for the application. The text and graphic may be used together or separately. The text font and size may also be adjusted as required.

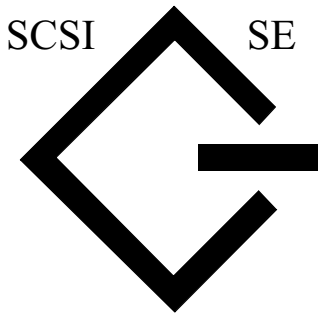


Figure K.1 - SE icon for SCSI

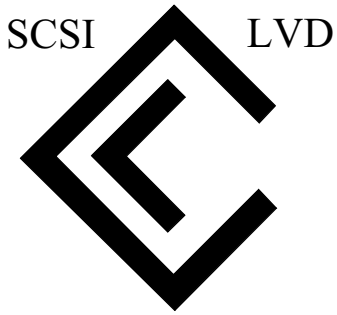


Figure K.2 - LVD icon for SCSI

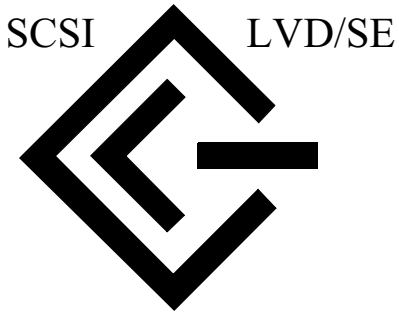


Figure K.3 - SE/LVD multimode icon for SCSI

Annex L

(informative)

Backplane construction guidelines

L.1 Universal backplane construction

L.1.1 Universal backplane construction overview

Printed circuit boards are constructed using either microstrip or stripline or a combination of both for routing signals. The important electrical characteristics may be determined from the geometry and the material properties of the microstrip or stripline.

L.1.2 Microstrip

See figure L.1 for the microstrip geometry.

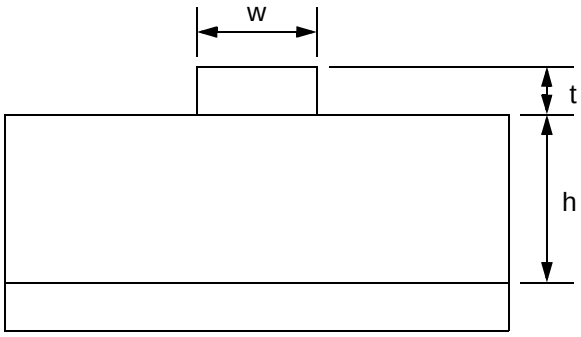


Figure L.1 - Microstrip geometry

The characteristic impedance (Z_0) for the microstrip geometry shown in figure L.1 is defined by the following equation:

$$Z_0 = \frac{87}{(e_r + 1,41)^{1/2}} \times \ln\left(\frac{5,98h}{0,83w + t}\right)\Omega$$

The propagation delay (T_{pd}) for the microstrip geometry shown in figure L.1 is defined in the following equation:

$$T_{pd} = 33,36(0,475e_r + 0,67)^{1/2} \quad \text{ps/cm}$$

Where e_r is equivalent to the relative dielectric constant.

L.1.3 Embedded Microstrip

See figure L.2 for the embedded microstrip geometry.

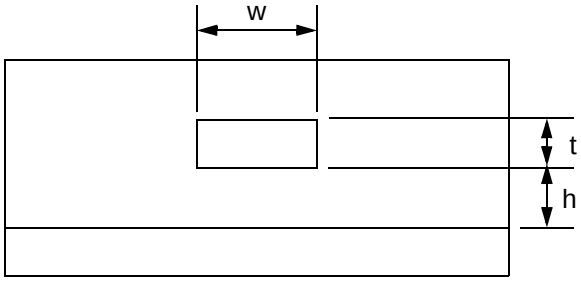


Figure L.2 - Embedded microstrip geometry

The characteristic impedance (Z_0) for the embedded microstrip geometry shown in figure L.2 is defined by the following equation:

$$Z_0 = \frac{K}{(0,805e_r + 2)^{1/2}} \times \ln\left(\frac{5,98h}{0,8w + t}\right)\Omega$$

The propagation delay (T_{pd}) for the microstrip geometry shown in figure L.2 is defined by the following equation:

$$T_{pd} = 33,36(0,475e_r + 0,67)^{1/2} \quad \text{ps/cm}$$

Where:

a) e_r is equivalent to the relative permittivity; and
b) K is a constant between 60 and 65 that varies with the thickness of the dielectric covering the conductor. If the thickness is below 15 mils K = 65 and at 20 mils K = 60.

L.1.4 Stripline

See figure L.3 for the stripline geometry.

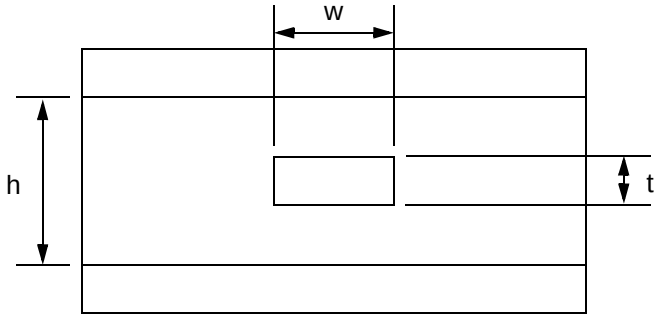


Figure L.3 - Stripline geometry

The characteristic impedance (Z_0) for the stripline geometry shown in figure L.3 is defined by the following

equation:

$$Z_0 = \frac{60}{(e_r)^{1/2}} \times \ln\left(\frac{4h}{0,67\pi w(0,8 + \frac{t}{w})}\right)\Omega$$

The propagation delay (T_{pd}) for the stripline shown in figure L.3 is defined by the following equation:

$$T_{pd} = 33,36(e_r)^{1/2} \text{ ps/cm}$$

Where e_r is equivalent to the relative permittivity.

L.1.5 Dual Stripline

See figure L.4 for the dual stripline geometry.

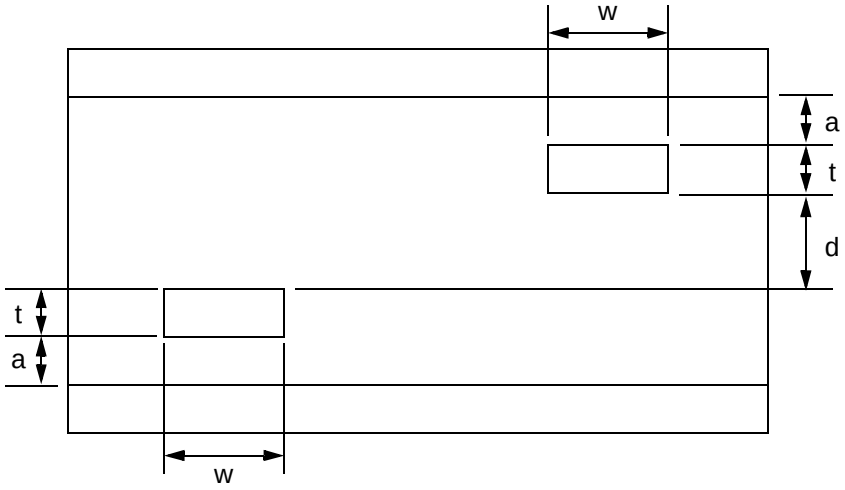


Figure L.4 - Dual stripline geometry

The characteristic impedance (Z_0) for the dual stripline geometry shown in figure L.4 is defined by the following equations:

$$Z_0 = \left(\frac{2F_1F_2}{F_1 + F_2}\right)\Omega$$

Where:

a) $F_1 = \frac{60}{(e_r)^{1/2}} \times \ln\left(\frac{8a}{0,67\pi w(0,8 + \frac{t}{w})}\right)\Omega$; and

b) $F_2 = \frac{60}{(e_r)^{1/2}} \times \ln\left(\frac{8(a+d)}{0,67\pi w(0,8+\frac{t}{w})}\right)\Omega$.

The propagation delay (T_{pd}) for the stripline shown in figure L.3 is defined by the following equation:

$$T_{pd} = 33,36(e_r)^{1/2} \text{ ps/cm}$$

Where e_r is equivalent to the relative permittivity.

L.1.6 Differential Impedance

The lossless model (i.e., doesn't include shunt conductance and series resistance) for the differential transmission line contains a series inductance, mutual inductance, capacitance to ground and a differential capacitance (see figure L.5).

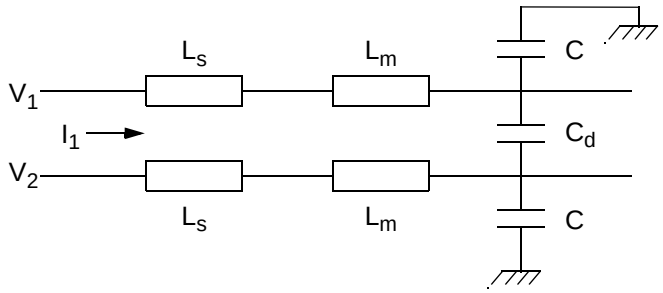


Figure L.5 - Lossless model for differential impedance

The two conductors are symmetrical therefore the two self inductance's are equal and the two capacitors to ground are equal ($Z_{11} = Z_{21}$). The voltage applied between the two lines may be thought of as a superposition of two voltages, a common mode voltage V_c (even mode) and a differential mode voltage V_d (odd mode) where: $V_c = (V_1+V_2)/2$ and $V_d = (V_1-V_2)/2$.

When a differential voltage is applied, equal and opposite currents flow and the voltage drop across the mutual inductance is in a direction opposite that of the self inductance. In the differential voltage case the mutual inductance acts to reduce the inductance seen by the differential mode signal to L_s - L_m. The signals are moving in opposite directions therefore the effect of the mutual capacitance C_d is doubled. The total capacitance seen by the differential signal is C + C_d. For this case the odd mode impedance (Z_d) is calculated by the following equation:

$$Z_d = \frac{V_d}{I_1} = \left(\frac{L_s - L_m}{C + C_d}\right)^{1/2} \Omega$$

When a common mode voltage source drives the bus the current flows in the same direction the odd mode impedance (Z_c) is calculated by the following equation:

$$Z_c = \frac{V_c}{I_1} = \left(\frac{L_s + L_m}{C - C_d}\right)^{1/2} \Omega$$

The differential mode impedance is two times the odd mode impedance ($Z_{diff} = 2Z_d$).

The differential impedance may also be derived from the matrix of a two port network as shown in the following equation:

$$\begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{bmatrix} \times \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}$$

Because in differential mode $I_1 = -I_2$ the matrix of a two port network the differential impedance (Z_d) is calculated by the following equation:

$$Z_d = \frac{V_d}{I_1} = \frac{V_1 - V_2}{I_1}\Omega$$

Therefore, from the matrix for the two port network the following equation is derived:

$$\frac{V_1 - V_2}{I_1} = (Z_{11} + Z_{22}) - (Z_{12} - Z_{21})\Omega$$

For a passive black box $Z_{12} = Z_{21}$. For a symmetrical system $Z_{11} = Z_{21}$. Therefore the differential impedance may be defined as $Z_{diff} = 2(Z_{11} - Z_{12})\Omega$ and the common mode impedance may be defined as $Z_{cm} = 2(Z_{11} + Z_{12})\Omega$.

If the traces are loosely coupled then Z_{12} is negligible.

L.1.7 Single ended impedance

The odd mode impedance approaches the single ended impedance as the trace spacing becomes larger than the height above the ground plane. A practical rule for zero coupling between traces is; if the trace to trace spacing is three times the trace to ground plane spacing then the trace to trace coupling is negligible.

If V_2 in figure L.5 is grounded then the single ended impedance is determined by the self inductance and the sum of the differential capacitance plus the single ended capacitance. In this case the single ended impedance (Z_o) is calculated by the following equation:

$$Z_o = \left(\frac{L_s}{C}\right)^{1/2}\Omega$$

L.1.8 Differential stripline

For differential stripline there are two choices for routing, edge coupled (side by side) figure L.7 and figure L.7, and broadside coupled (stacked) figure L.8.

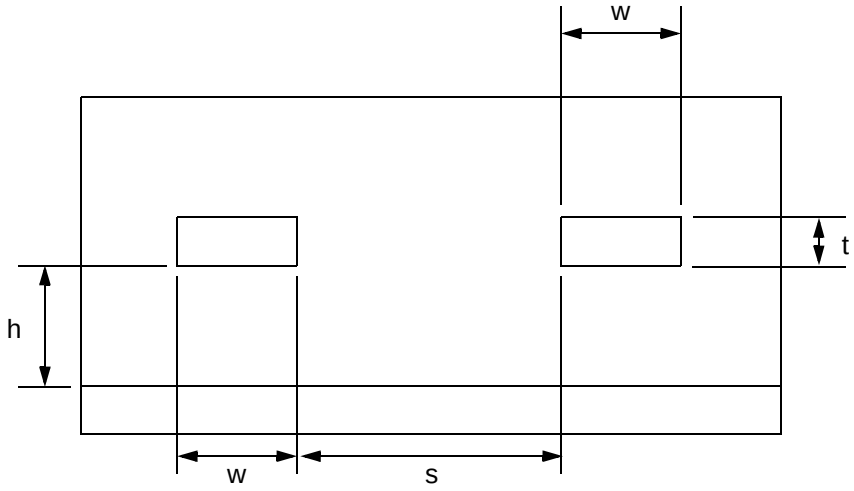


Figure L.6 - Edge coupled differential microstrip

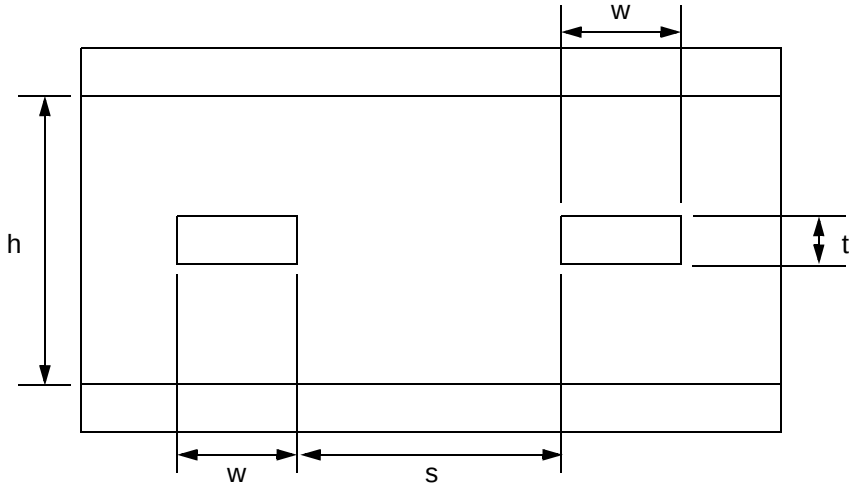


Figure L.7 - Edge coupled differential stripline

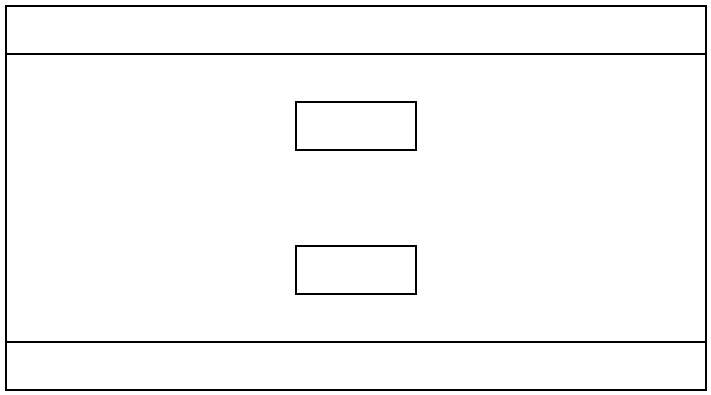


Figure L.8 - Broadside coupled differential stripline

Edge coupled is the most common. Broadside has better coupling characteristics but may be difficult to bring to a common plane and requires a thicker dielectric to keep from coupling to one of the reference planes. Therefore this annex does not define the broadside coupled differential stripline impedance characteristics.

The differential impedance is determined primarily by the conductor spacing and dielectric thickness. First the SE impedance (Z_0) is calculated (see L.1.7) then it is used along with conductor spacing and dielectric thickness to calculate the differential impedance. The calculations assume the traces of the differential pairs have the same physical cross section dimensions.

The practical range of Z_0 and Z_{diff} is from 20 Ohms to 150 Ohms with a typical range being between 50 Ohms and 110 Ohms. This impedance depends on the trace width and distance to ground. If the trace is wide and close to ground it is more capacitive and has a lower impedance. If the trace is narrow and the spacing from the ground plane is large, the trace is more inductive and has a higher impedance. Controlled impedance boards with the impedance's matching within several Ohms usually have a characteristic impedance in the 50 Ohm to 80 Ohm range. This is due to manufacturing constraints such as maximum dielectric thickness and minimum trace widths.

Using figure L.6 as an example the edge coupled microstrip differential impedance (Z_{diff}) is calculated by the following equation:

$$Z_{diff} = 2Z_0(1 - 0,48e^{(-(0,96s)/h)})\Omega$$

Using figure L.7 as an example the edge coupled stripline differential impedance (Z_{diff}) is calculated by the following equation:

$$Z_{diff} = 2Z_0(1 - 0,374e^{(-2,9s/h)})\Omega$$

The trace capacitance and inductance are important parameters for calculating the transmission line parameters. The method used to calculate these parameters is shown below.

The edge coupled microstrip trace capacitance (C_0) and inductance (L_0) are calculated by the following equations:

$$C_0 = \frac{T_d}{Z_0} = \frac{33,36(0,475e_r + 0,67)^{1/2}}{Z_0} \text{ pF/cm}$$

$$L_0 = Z_0T_d = 33,36Z_0(0,475e_r + 0,67)^{1/2} \times 10^{-3} \text{ nH/cm}$$

Where:

a) T_d is the propagation delay; and
b) e_r is equivalent to the relative permittivity.

The edge coupled stripline trace capacitance (C_0) and inductance (L_0) are calculated by the following equations:

$$C_0 = \frac{T_d}{Z_0} = \frac{33,36(e_r)^{1/2}}{Z_0} \qquad \text{pF/cm}$$

$$L_0 = Z_0 T_d = 33,36 Z_0 (e_r)^{1/2} \times 10^{-3} \quad \text{nH/cm}$$

Where:

a) T_d is the propagation delay; and
b) e_r is equivalent to the relative permittivity.

As demonstrated above the characteristic impedance for the backplane is primarily determined by:

a) Width and thickness of the conductors;
b) dielectric constant of the substrate material; and
c) the substrate material thickness between the conductor and reference planes.

L.1.9 Dielectric material selection

Proper selection of the dielectric material is very important for high speed PCB's. Two key parameters are the dielectric constant of the material and the loss tangent. The dielectric constant relates to the material's ability to hold charge and the loss tangent refers to how much of the energy is lost in the material due to dissipation. The ideal materials have small numbers. Table M.1 gives a sample of some materials.

Table L.1 - Dielectric constants

Material	Dielectric Constant	Loss Tangent
Air	1,0	0
PTFE (teflon)	2,1 - 2,5	0,0002 - 0,002
BT resin	2,9 - 3,9	0,003 - 0,12
Polyimide	2,8 - 3,5	0,004 - 0,02
Silica	3,8 - 4,2	0,0006 - 0,005
Polyimide/Glass	3,8 - 4,5	0,003 - 0,01
Epoxy/Glass (FR-4)	4,1 - 5,3	0,002 - 0,02

The most frequently used dielectric are a glass-epoxy (G-10) and a derivative, FR-4. The FR-4 material has acceptable performance for differential.

For higher speeds, materials like teflon should be considered, although they are much more expensive. PCB manufacturers publish specifications with their boards, among them should be the dielectric constant, loss tangent, and other electrical properties. The board tolerances should also be specified. For example in a FR-4 PCB the dielectric constant may change by as much as 10 % on a single board. These changes affect the propagation velocity and lead to skew.

L.1.10 Vias

Vias have a parasitic capacitance and inductance that may affect signal quality. The size and density of vias also affects how many traces that may be run between them on inner layers.

The parasitic capacitance (C) to ground of a via is calculated by the following equation:

$$C = \frac{3,58 e_r T D_1}{D_2 - D_1} \text{ pF}$$

Where:

a) D_2 is the diameter of the clearance hole in centimeters;
b) D_1 is the diameter of pad surrounding the via in centimeters; and
c) T is the thickness of the printed circuit board in centimeters.

A typical via capacitance is in the 0,2 pF to 0,8 pF range. For critical differential signals the via mismatch should be kept to a minimum.

The parasitic inductance (L) of a via is calculated by the following equation:

$$L = (12,9h)\left(\ln\left(\frac{4h}{d}\right) + 1\right) \text{ nH}$$

Where:

a) h is the length of the via in centimeters; and
b) d is the diameter of the via in centimeters.

The via inductance is most detrimental when they are in series with termination power bypass capacitors.

The goal of the universal backplane is to have the SE and LVD backplane impedance to match the SE and LVD backplane impedance of the cable. One of the aspects of this is to determine if the PCB traces are long enough to require the traces to be treated as transmission lines. If the signal electrical length is greater than 1/2 of the rising edge (T_{rise}), then the PCB exhibits transmission line effects. This length may be expressed as:

$$\text{Length} = \frac{T_{rise}}{2(LC)^{1/2}} \text{ cm}$$

If the round trip time for the switching waveform is greater than the rise or fall time of the driving SCSI device, the settling of the transmission line effects are not hidden during the rise and fall time of the driving SCSI device. In other words, in order to be a transmission line $2xT_{pd} > T_{rise}$ or T_{fall} (the minimum of the two) where T_{pd} is the one way propagation delay.

Specifically for FR-4 the two following formulas may be used.

a) BW>236/d where d is in centimeters and the bandwidth is BW=0,35/T_{rise}; and
b) F_{clock}>47/d.

Complete formulas for the maximum length without termination or controlled impedance may also be used.

The microstrip maximum length ($Length_{max}$) is calculated as shown in the following equation:

$$\text{Length}_{max} = \frac{((C + Z_0)^2 + (12,3 \times 10^6))^{1/2} - (C_t Z_0)}{66,7(0,475 e_r + 0,67)^{1/2}} \text{ cm}$$

The stripline maximum length ($Length_{max}$) is calculated as shown in the following equation:

$$Length_{max} = \frac{((C + Z_0)^2 + (12,3 \times 10^6))^{1/2} - (C_t Z_0)}{66,7(e_r)^{1/2}} \text{ cm}$$

Where:

a) Z_0 is the unloaded impedance;
b) C_t is the total load capacitance;
c) e_r is equivalent to the relative permittivity; and
d) C is the unloaded characteristic capacitance.

Depending on how conservative the calculation is PCB lengths in the four inch range or greater should probably be treated as transmission lines for differential risetimes in the 1 ns range.

L.1.11 Transmission Lines

A typical transmission line element (figure L.9) is broken into four parts, a series resistance, series inductance, shunt conductance, and shunt capacitance.

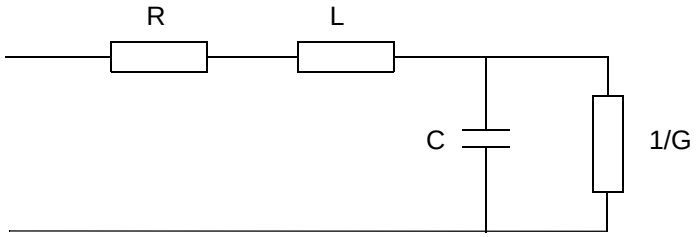


Figure L.9 - Typical transmission line element

When using this model the minimum wavelength should be much longer than the individual section to make it appear as a distributed model. The non-linearities are usually ignored. The general form for the impedance (Z_0) is shown in the following equations:

$$Z_0 = \left(\frac{l(R + jwL)}{2} + \frac{1}{2}\left(l^2(R + jwL)^2 + 4\left(\frac{R + jwL}{G + jwC}\right)\right)^{1/2}\right)\Omega$$

or

$$Z_0 = \left(\frac{l(R + jwL)}{2} - \frac{1}{2}\left(l^2(R + jwL)^2 + 4\left(\frac{R + jwL}{G + jwC}\right)\right)^{1/2}\right)\Omega$$

Where:

a) *j* denotes the complex number j x j = -1;
b) *w* is omega or 2 x pi x frequency; and
c) *l* is the section length in centimeters.

If you make the section length (*l*) of the segment small enough, the result is the distributed model. Where the impedance (Z_0) is calculated as shown in the following equation:

$$Z_0 = \left(\frac{R + jwL}{G + jwC}\right)^{1/2} \Omega$$

If the transmission rate is high (i.e., w/2π > 100 KHz) then *w*L and *w*C are much larger than R and G and the impedance becomes the more commonly used form of; $Z_0 = (L/C)^{1/2}\Omega$.

The other extreme is if w/2π ≤ 1 KHz then; $Z_0 = (R/G)^{1/2}\Omega$.

Another important factor is the propagation velocity and propagation delay. The propagation delay per unit length (T_{delay}) is calculated as shown in the following equation;

$$T_{delay} = \frac{T_1}{l}[(R + jwL) + (G + jwC)]^{1/2} \text{ nsec/cm}$$

However, if the line is short then T_{delay} is calculated as shown in the following equation:

$$T_{delay} = jwl(LC)^{1/2} \text{ nsec}$$

Therefore the propagation velocity is $V = l/(LC)^{1/2}$ cm/nsec for lossless lines.

The time delay is $T = 1/V = (LC)^{1/2}$ nsec that gives a total propagation delay of $T = l(LC)^{1/2}$ nsec/cm. The two equations most commonly used for hand calculations are:

a) for characteristic impedance: $Z = (L/C)^{1/2}\Omega$; and
b) for propagation delay: $T = (LC)^{1/2}$ nsec/cm.

If these values are not specified they may be calculated from the cross section geometry and dielectric material used for the PCB. The equation using the geometry and material are based on the above equation, but should also be treated as if they were operating in the transverse electromagnetic mode.

Another factor that should be taken into account is attenuation due to the skin effect. The skin depth is;

$$\delta = 500(\rho/f)^{1/2} \text{ meter.}$$

Where:

a) ρ resistivity in Ohm-meter;
b) f frequency in hertz; and
c) δ thickness in meters.

The low loss attenuation per length is; $\alpha = 4,34\left(\frac{R_1}{Z_0} + G_1 Z_0\right)$ dB/length.

The loss due to the dielectric is; $\alpha = 0,9f(\tan(\delta) \times e_r^{1/2})$ dB/cm where f is in GHz.

The loss due to the metal for t<δ is; $\alpha = 17,1\rho/(twZ_0)$ dB/cm where:

a) ρ resistivity in Ohm-meter;
b) *t* thickness in microns;
c) *w* the width in microns; and
d) Z_0 the single ended impedance.

The loss due to the metal for t>δ is; $\alpha = \frac{0,22(\rho f)^{1/2}}{w}$ dB/cm where:

a) f frequency in GHz;
b) ρ resistivity in Ohm-meter; and
c) *w* the width in microns.

When calculating the impedance and propagation delay, the capacitive loads have to be accounted for. Capacitive loading decreases the impedance and increases the propagation delay.

The loaded propagation delay (T'_{pd}) is calculated as shown in the following equation:

$$T'_{pd} = T_{pd}\left(1 + \left(\frac{C_D}{C_0}\right)\right)^{1/2} \text{ nsec}$$

or

$$T'_{pd} = (L(C_D + C_0))^{1/2} \text{nsec}$$

The loaded characteristic impedance (Z'_0) is calculated as shown in the following equation:

$$Z'_0 = \frac{Z_0}{\left(1 + \left(\frac{C_D}{C_0}\right)\right)^{1/2}} \Omega$$

or

$$Z'_0 = \left(\frac{L}{C_D + C_0}\right)^{1/2} \Omega$$

Where:

a) C_0 is the intrinsic capacitance;
b) C_D is the load capacitance;
c) Z_0 is the single ended impedance; and
d) T_{pd} is the propagation delay.

Besides reducing the impedance and increasing the propagation delay, a heavily loaded trace also slows the rise and fall times of the drivers and filters (RC filter) out some high frequency components. The loaded propagation delay (T'_{pd}) should be used when deciding whether or not to treat the trace as a transmission line.

When adding the load capacitance; sockets, connectors, vias, and IC's add to the distributed capacitance. Using traces with a higher intrinsic capacitance reduce the effects of the loading (e.g., microstrip is faster

than stripline, but is affected more by loading since it has a lower characteristic capacitance.)

The complete forms of loaded propagation delay (T'_{pd}) and impedance (Z'_0) for microstrip are calculated as shown in the following equations:

$$T'_{pd} = 5,776(15,85e_r + 25,35 + C_D Z_0(0,475e_r + 0,67)^{1/2})^{1/2} \text{ ps/cm}$$

$$Z'_0 = \frac{Z_0}{\left(\frac{C_D Z_0}{33,36(0,475e_r + 0,67)^{1/2}} + 1\right)^{1/2}} \Omega$$

The complete forms of loaded propagation delay (T'_{pd}) and impedance (Z'_0) for stripline are calculated as shown in the following equations:

$$T'_{pd} = 5,776(33,36e_r + C_D Z_0(e_r)^{1/2})^{1/2} \text{ ps/cm}$$

$$Z'_0 = \frac{Z_0}{\left(\frac{C_D Z_0}{33,36(e_r)^{1/2}} + 1\right)^{1/2}} \Omega$$

The impedance mismatches between loads, sources, connectors, cables, and traces may cause transmission line effects such as ringing, stair step effects, and long bus settle delays. These are caused by reflections at the impedance discontinuities, the reflection coefficients are:

a) at load: $\Gamma_1 = \frac{Z_1 - Z_0}{Z_1 + Z_0}$ and

b) at source $\Gamma_s = \frac{Z_s - Z_0}{Z_s + Z_0}$.

These reflections, if large enough, may cause false transitions. They may also cause standing waves.

$$VSWR = \frac{1 + \Gamma}{1 - \Gamma}$$

Where VSWR is the voltage standing wave ratio.

The standing waves are at quarter wavelengths (λ/4) where the wavelength is calculated by the following equation:

$$\lambda = \frac{1}{f(LC)^{1/2}}$$

To capture all the harmonic content of the square wave the frequency used should be $f=0,5/t_{rise}$. The signal's magnitude at time t is $t = (\Gamma_1 \Gamma_s)^{t/T}$ where T is the one way propagation delay.

It is important to remember that the driver rise and fall times are very important in the behavior of the transmission line and that reflections may cause long bus settle times.

Some general guidelines for laying out PCB's are:

a) Evenly distribute loads along the trace, this gives a distributed load and reduces reflections from discontinuities;
b) avoid T's for critical signals;
c) add sockets and vias into capacitance calculations;
d) keep trace lengths as short as possible;
e) use as low a dielectric constant material as possible;
f) if possible use a controlled impedance PCB, so that the behavior is predictable;
g) balance path lengths to reduce skew; and
h) minimize length through the connector.

Crosstalk is always an issue and all TTL/CMOS signal paths should be isolated from differential signal paths. Since crosstalk is proportional to dv/dt crosstalk may easily occur if lines with large voltage swings are near differential lines. To isolate the lines, either increase the separation, run ground traces between them, or isolate them by using different planes. Since crosstalk is caused by the capacitive coupling between signals and the mutual inductance some general observations are:

a) Crosstalk scales with signal amplitude;
b) crosstalk is proportional to slew rate (Voltage magnitude of output waveform/T_{rise}) with slower rise/fall times yielding less crosstalk; and
c) far end crosstalk width is equivalent to the rise/fall time. The crosstalk width is related to the slew rate in b). The faster the slew rate the larger the crosstalk

Annex M

(informative)

SPI-3 to SCSI-2 terminology mapping

This annex contains a mapping of terminology used in SCSI-2 to the terminology used in this standard (see table M.1).

Table M.1 - SPI-3 to SCSI-2 terminology mapping

SPI-3 equivalent term	SCSI-2 term
abort task	abort tag
abort task set	abort
cable skew	cable skew delay
clear task set	clear queue
head of queue	head of queue tag
ordered	ordered queue tag
simple	simple queue tag
target reset	bus device reset
task	I/O process
task complete	command complete
task set	queue

Annex N

(informative)

Physical layer integrity checking

N.1 Introduction

N.1.1 Overview and glossary

This annex defines integrity checking and fall back terminology.

Tools used to perform integrity checking include:

a) the INQUIRY command;
b) enhancements to the READ BUFFER and WRITE BUFFER commands; and
c) the synchronous transfer timeout function in the port control mode page (see 18.1.4).

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

N.1.2 Integrity checking: The act of verifying that the physical layer is able to transfer test data at the negotiated speed and width between the initiator and target (i.e., a quick check for physical domain validation). For example, two wide SCSI devices connected with a narrow cable will discover that the cable does not support wide transfers during this checking. These SCSI devices will then re-negotiate to narrow transfers.

N.1.3 Fall back: The act of re-negotiating to a set of physical parameters that are less demanding. After falling back, integrity checking is again performed to verify the new parameters. This cycle may be repeated until an acceptable set of physical parameters is found.

N.2 Integrity checking methods

N.2.1 Basic integrity check

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

This test detects most basic problems including:

a) Path width errors (i.e., narrow cable used with wide SCSI devices);
b) expander errors (e.g., expanders not capable of the negotiated data rate);
c) gross cable errors (e.g., broken wire);
d) incorrect termination (e.g., missing or bad terminator); or
e) damaged transceiver.

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

N.2.2 Enhanced integrity check

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

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

a) Counting (0001h, 0203h, 0405h, …);
b) alternating ones and zeros (0000h, FFFFh, 0000h, FFFFh, …);
c) crosstalk (5555h, AAAAh, 5555h, AAAAh, …); and
d) shifting bit (0000h, FFFEh, 0000h, FFFDh, … then FFFFh, 0001h, FFFFh, 0002h, …).

This test may detect additional problems including:

a) Wrong impedance cables;
b) bad SCSI device spacing;
c) poor termination;
d) marginal transceivers;
e) excessive crosstalk; or
f) excessive system noise.

N.3 Fall back

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

NOTE 66 - If two devices conforming to this standard are connected through a simple expander that was design to operate with devices conforming to a prior SCSI standard, then the simple expander may not operate correctly if the fall back negotiation relies on PPR messages. It is recommended that the fall back procedure first negotiate asynchronous mode using PPR messages, then negotiate the fall back mode using WDTR/SDTR messages.

N.4 System considerations

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

N.5 Buffer protection

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

The READ BUFFER and WRITE BUFFER commands include an echo buffer option that provides buffer protection in multi-initiator environments. Other mechanisms that may help prevent buffer corruption in multiple initiator environments are RESERVE/RELEASE and linked commands.

N.6 Failure modes during integrity checking

Integrity checking may cause several kinds of error conditions:

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

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

a) Assert the RST signal, or
b) use the synchronous transfer timeout (see 18.1.4) in the SCSI port control mode page.

Annex O

(informative)

Alternate error detection for the asynchronous information phases (COMMAND, MESSAGE, and STATUS)

O.1 Error detection for asynchronous information phases

This annex describes an enhanced error detection method for the COMMAND, MESSAGE, and STATUS asynchronous information transfer phases. In systems not implementing this scheme, these phases only transfer information on the lower eight data bits of a SCSI bus with only normal parity protection on those transfers. Therefore, additional check information may be transferred on the upper eight data bits in order to improve error detection capabilities. Since the upper eight data bits of the bus are used for this scheme, this error detection method is only available on wide SCSI devices that are on wide SCSI busses.

O.2 Protection code

O.2.1 Protection code overview

The following are the covered signals to be encoded and details of the protection code to be used on the asynchronous information phases.

O.2.2 Covered signals

Table O.1 defines the signals to be covered by the protection code and their bit locations in the 21-bit code word. When a SCSI device receives an information byte, it also latches the state of the other SCSI signals and values noted in table O.1.

Table O.1 - Signals be covered by the protection code and their bit locations

Code Word Bit Location	SCSI Signal	Meaning
0	DB(0)	Data bit 0 of the information byte
1	DB(1)	Data bit 1 of the information byte
2	DB(2)	Data bit 2 of the information byte
3	DB(3)	Data bit 3 of the information byte
4	DB(4)	Data bit 4 of the information byte
5	DB(5)	Data bit 5 of the information byte
6	DB(6)	Data bit 6 of the information byte
7	DB(7)	Data bit 7 of the information byte
8	DB(8)	Reserved (see note 1)
9	DB(9)	Reserved (see note 1)
10	RSVD	Reserved (see note 2)
11	RSVD	Reserved (see note 2)
12	RSVD	Reserved (see note 2)
13	Seq ID 0	Sequence ID bit 0
14	Seq ID 1	Sequence ID bit 1
15	DB(10)	Redundant bit 0 of the code word
16	DB(11)	Redundant bit 1 of the code word
17	DB(12)	Redundant bit 2 of the code word
18	DB(13)	Redundant bit 3 of the code word
19	DB(14)	Redundant bit 4 of the code word
20	DB(15)	Redundant bit 5 of the code word

Note:
1 DB(8) and DB(9) are reserved for future use. These signals are negated by the transmitting SCSI device and are ignored by the receiving SCSI device. Both the transmitter and receiver encode these signals in the protection code.
2 For calculation purposes these signals are zero. However, these virtual signals may be used for other functions in a future standard.

The Sequence IDs are encoded in the protection code. A sequence of consecutive information transfers during a MESSAGE, COMMAND, or STATUS phase is a run. The sequence ID increments during a run. A new run begins on every phase change or on each MESSAGE OUT retry.

For each new run, the sequence ID is set to zero for the first word transferred, set to one for the second word transferred, set to two for the third word transferred, and set to three for the fourth word transferred. The sequence ID then cycles back to being set to zero for the fifth word transferred, and so forth until the run is complete. At the beginning of the next run, the sequence ID is set to zero again.

The sequence ID provides detection of errors that occur when an information transfer is missed or double clocked. A sequence ID error causes a protection code error. If a protection code error is detected, then the information transfer is invalid. The method for recovery from these errors is the same as the method for parity error recovery.

O.2.3 Code description

The protection code is a cyclic binary Bose, Chaudhuri and Hocquenghem (BCH) code.

Table O.2 - Protection code

Code	Maximum data bits allowed	Number of redundant bits	Minimum distance of the code
(21,15,4)	15	6	4

The BCH protection code is a cyclic code with a generator polynomial of $x^6 + x^5 + x^2 + 1$.

The canonical form of the code generator is shown in figure O.1. This is a serial implementation: the register is initialized to zero, then the data is fed in one bit at a time, codeword bit 14 (as defined above) first, followed by codeword bits 13, 12, 11, and so on until bit 0. As each data bit is input, the shift register is clocked. When all 15 bits have been clocked into the generator, the check bits are available in the registers, check bit 0 (codeword bit 15) on the right in the diagram and check bit 5 (codeword bit 20) on the left. The **+** signs represent an XOR operation.

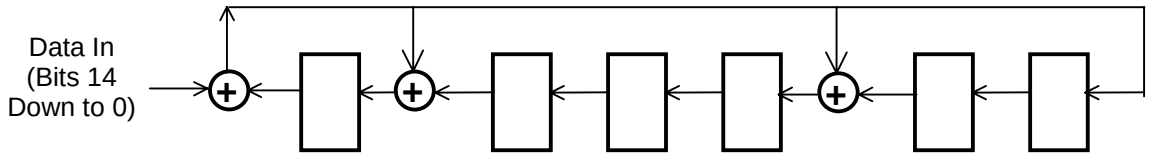


Figure O.1 - Protection code generator

Using this representation as a baseline, it is possible to construct logic to generate the six check bits from an input data stream of n-bit width, including all 15 bits simultaneously, which is the expected implementation, wherein:

Redundant bit 0 is the XOR of code word bits 0, 1, 2, 3, 5, 6, 7, 10, 11, and 13.
Redundant bit 1 is the XOR of code word bits 1, 2, 3, 4, 6, 7, 8, 11, 12, and 14.
Redundant bit 2 is the XOR of code word bits 0, 1, 4, 6, 8, 9, 10, 11, and 12.
Redundant bit 3 is the XOR of code word bits 1, 2, 5, 7, 9, 10, 11, 12, and 13.
Redundant bit 4 is the XOR of code word bits 2, 3, 6, 8, 10, 11, 12, 13, and 14.
Redundant bit 5 is the XOR of code word bits 0, 1, 2, 4, 5, 6, 9, 10, 12, and 14.

See O.6 for coding examples and examples of protection coding output.

O.2.4 Error detection properties

This protection code was selected to have adequate detection properties for asynchronous information transfer phases, given that these transfers are inherently less prone to errors and that these transfers have short code words (approximately 20 bits as compared to thousands of bits during a DT data phase). The

BCH protection code Hamming distance is a minimum of four, the same as achieved by the data CRC for transfers of less than eight kilobytes. The protection code will detect all errors of three bits or fewer, all errors of an odd number of bits, and 98,4 % of all potential errors.

O.3 Protection code usage

O.3.1 Protection code usage overview

Protection code checking is enabled or disabled on an I_T nexus basis. All COMMAND, MESSAGE, and STATUS phase information is checked for an I_T nexus while checking is enabled. Protection code checking is disabled after a power cycle, after a hard reset, after a TARGET RESET message, and after a change in the transceiver mode (e.g., LVD mode to MSE mode). Protection code checking is always disabled for information unit transfers.

O.3.2 Protection code transmission

SCSI devices supporting this protection code transmit the protection code check data during all COMMAND, MESSAGE, and STATUS phases. The protection code byte is transferred on the upper eight bits of a wide bus simultaneously with the information day byte on the lower eight bits of the bus using the same clock for the transfer. Thus the transfer of the information byte and the protection code byte is performed exactly like a normal wide transfer. The check data is transmitted even if detection is not enabled.

O.3.3 Enabling protection code checking

A SCSI device enables protection code checking for an I_T nexus when it detects that valid protection code data is being transmitted on the upper byte of the SCSI bus. The frequency that a SCSI device will try to enable protection code checking and the number of valid protection code bytes required is vendor specific. The following are some potential times when a SCSI device may try to enable protection code checking:

a) During the first COMMAND, MESSAGE, or STATUS phase after a power cycle, after a hard reset, after a TARGET RESET message, or after a change in the transceiver mode.
b) Any time that removal and insertion of a SCSI device is possible, i.e. after a UNIT ATTENTION condition.
c) During the MESSAGE phases of a negotiation.

O.3.4 Disabling protection code checking

The removal and insertion of a SCSI device may require that protection code checking be disabled for a previously enabled I_T nexus. A SCSI device disables protection code checking when it detects that no protection code data is being transmitted on the upper byte. The determination that no protection code data is being transmitted is vendor specific. The following are some potential ways that a SCSI device may determine that no protection code data is being transmitted:

a) The DB(15-8) and DB(P1) signals are continuously deasserted while there is good parity on DB(7-0) and DB(P_CRCA).
b) The protection code has a consistent error while there is good parity on DB(7-0) and DB(P_CRCA).

O.4 Parity

When protection code checking is enabled normal wide parity is used during a protected transfer of COMMAND, MESSAGE, or STATUS information. DB(P_CRCA) contains the parity for DB(7-0), and DB(P1) contains the parity for DB(15-8).

O.5 Error handling

Protection code errors are handled exactly like parity errors during COMMAND, MESSAGE, or STATUS phases as defined in the relevant subclauses on exception condition handling in 11.1.

O.6 Examples

O.6.1 C code example

The following is an example of a program written in the C programming language that would generate the check bits for the protection scheme described in this annex.

```
/* C-code implementation of (21,15,4) cyclic code calculated in parallel.   */
/* Input a 15-bit word, output six check bits.                               */
/* The implementation splits the 15-bit word into an 8-bit and 7-bit word.  */
/* Each word is input to its own lookup table, producing two 6-bit results  */
/* The two 6-bit results are then XOR'd together to create the final six    */
/* check bits.                                                               */
/* The correspondence between the input data and SCSI bus is:                */
/* data[0]  :   SCSI DB[0]                                                   */
/* data[1]  :   SCSI DB[1]                                                   */
/* data[2]  :   SCSI DB[2]                                                   */
/* data[3]  :   SCSI DB[3]                                                   */
/* data[4]  :   SCSI DB[4]                                                   */
/* data[5]  :   SCSI DB[5]                                                   */
/* data[6]  :   SCSI DB[6]                                                   */
/* data[7]  :   SCSI DB[7]                                                   */
/* data[8]  :   SCSI DB[8]                                                   */
/* data[9]  :   SCSI DB[9]                                                   */
/* data[10] :   0 (Reserved)                                                 */
/* data[11] :   0 (Reserved)                                                 */
/* data[12] :   0 (Reserved)                                                 */
/* data[13] :   Sequence ID 0                                                */
/* data[14] :   Sequence ID 1                                                */

#include <stdio.h>
unsigned short gen_poly = 0145U;
unsigned short degree_term = 0100U;
unsigned short table_lower[256];
unsigned short table_upper[128];

unsigned short encode( unsigned short data );
void making_tables( void );

void main( void ) {
   unsigned short data = 1U;

   making_tables();

   /* to exit the loop, enter 0.  Data of 0 produces result of 0 */

   while( data ) {
     /* read in an input data value */
     printf( "Enter a fifteen-bit data in <hex> format:\n" );
     scanf( "%hx",&data );
     data &= 0x7fffU; /** making sure that there are 15 bits in data **/
```

```
     printf( "data = <hex>%04x, parallel-encoded edc  = <hex>%02x\n",
             data, encode( data ));
   }
}

/* creates the lookup tables used to generate the check bits */
/* Look up table for low 8 bits of input data, 256 entries x 6 bits */
/* Look up table for high 7 bits of data, 128 entries x 6 bits */
void making_tables( void ) {
   unsigned int k;
   unsigned int power_of_two, one;

   /** making table for the lower order 8 bits of data ***/
   for( k = 0U; k < 256U; k++ )
      table_lower[k] = 0U;

   table_lower[1] = gen_poly ^ degree_term;
   power_of_two = 1U;

   for( k = 1U; k < 8U; k++ ) {
      table_lower[power_of_two << 1] = table_lower[power_of_two] << 1;
      power_of_two <<= 1;

      if(table_lower[power_of_two] & degree_term)
         table_lower[power_of_two] ^= gen_poly;
   }

   for( k = 0U; k < 256U; k++ ) {
      if(table_lower[k])
         continue;

      for( one = 128U; one; one >>=1 ) {
         if( one & k )
            table_lower[k] ^= table_lower[one];
      }
   }

   /* May uncomment this loop to print out the table values */
   /* for( k = 0U; k < 256U; k++ ) {
         printf( "k = %03d, data = %04x\n", k, table_lower[k] );
      } */

   /** making table for the upper order 7 bits of data ****/
   for( k = 0U; k < 128U; k++ )
      table_upper[k] = 0U;

   table_upper[1] = table_lower[128] << 1;

   if( table_upper[1] & degree_term )
      table_upper[1] ^= gen_poly;

   power_of_two = 1U;

   for( k = 1U; k < 7U; k++ ) {
      table_upper[power_of_two << 1] = table_upper[power_of_two] << 1;
      power_of_two <<= 1;
```

```
      if( table_upper[power_of_two] & degree_term )
         table_upper[power_of_two] ^= gen_poly;
   }

   for( k = 0U; k < 128U; k++ ) {
      if( table_upper[k] )
         continue;

      for( one = 64U; one; one >>= 1 ) {
         if( one & k )
            table_upper[k] ^= table_upper[one];
      }
   }

   /* May uncomment this loop to print out the table values */
   /* for( k = 0U; k < 128U; k++ ) {
         printf( "k = %03d, data = %04x\n", k, table_upper[k] );
      } */
}

/* actual encoding is just two lookups and then XOR the results */
/** data contains 15 bits **/
unsigned short encode( unsigned short data ) {
   unsigned short edc, lower_index, upper_index;

   lower_index = data & 0xffU;          /** lower_index contains 8 bits **/
   upper_index = (data >> 8) & 0x7fU;   /** only 7 bits in upper_index **/
   edc = table_lower[lower_index] ^ table_upper[upper_index];

   return(edc);
}
```

O.6.2 Verilog® Hardware Description Language example

The following is an example of a Verilog® Hardware Description Language implementation that would generate the check bits for the protection scheme described in this annex.

```
// Verilog Implementation of protection code generator
//
//  Input:      cw          - 15-bit Code word
//  Returns:    check       - 6 Check bits
//
function[5:0]   check;
input[14:0]     cw;
begin
   check[0] = cw[00] ^ cw[01] ^ cw[02] ^ cw[03] ^ cw[05] ^ cw[06] ^ cw[07] ^ cw[10] ^ cw[11] ^ cw[13];
   check[1] = cw[01] ^ cw[02] ^ cw[03] ^ cw[04] ^ cw[06] ^ cw[07] ^ cw[08] ^ cw[11] ^ cw[12] ^ cw[14];
   check[2] = cw[00] ^ cw[01] ^ cw[04] ^ cw[06] ^ cw[08] ^ cw[09] ^ cw[10] ^ cw[11] ^ cw[12];
   check[3] = cw[01] ^ cw[02] ^ cw[05] ^ cw[07] ^ cw[09] ^ cw[10] ^ cw[11] ^ cw[12] ^ cw[13];
   check[4] = cw[02] ^ cw[03] ^ cw[06] ^ cw[08] ^ cw[10] ^ cw[11] ^ cw[12] ^ cw[13] ^ cw[14];
   check[5] = cw[00] ^ cw[01] ^ cw[02] ^ cw[04] ^ cw[05] ^ cw[06] ^ cw[09] ^ cw[10] ^ cw[12] ^ cw[14];
end
endfunction
```

O.6.3 Protection code examples

O.6.3.1 Message phase example

See table O.3 for an example of a sequence of an IDENTIFY message with a SIMPLE task attribute message having a tag field of zero

Table O.3 - Example of a sequence of an IDENTIFY message with a SIMPLE task attribute message having a tag field of zero

Information byte contents	Codeword bits (7:0) [DB(7:0)]	Codeword bits (9:8) [DB(9:8)]	Codeword bits (14:13) [Seq ID(1:0)]	Calculated redundant bits (5:0)	Output on DB(15:8)
IDENTIFY message	10000000	00	00	001011	00101100
SIMPLE message	00100000	00	01	110000	11000000
TAG = 0	00000000	00	10	110010	11001000

O.6.3.2 Command phase example

See table O.4 for an example of a sequence of a CDB for a READ(6) command with a logical block address of 1A BC DEh, and a transfer length of 55h.

Table O.4 - Example of a sequence of a CDB for a READ(6) command with a logical block address of 1A BC DEh, and a transfer length of 55h

Information byte contents	Codeword bits (7:0) [DB(7:0)]	Codeword bits (9:8) [DB(9:8)]	Codeword bits (14:13) [Seq ID(1:0)]	Calculated redundant bits (5:0)	Output on DB(15:8)
OPERATION CODE 08h (READ(6))	00001000	00	00	010011	01001100
LBA(20:16)	00011010	00	01	000011	00001100
LBA(15:8)	10111100	00	10	011110	01111000
LBA(7:0)	11011110	00	11	110110	11011000
transfer length	01010101	00	00	001111	00111100
control	00000000	00	01	011001	01100100

O.6.3.3 Example of a “shifting ones” sequence

Table O.5 is not a SCSI sequence, but demonstrates the calculation results and theoretical output for twelve different codewords each containing all zeroes except for a single one in a different bit in each word.

Table O.5 - Example of a “shifting ones” sequence

Information byte contents	Codeword bits (7:0) [DB(7:0)]	Codeword bits (9:8) [DB(9:8)]	Codeword bits (14:13) [Seq ID(1:0)]	Calculated redundant bits (5:0)	Theoretical output on DB(15:8)
Not applicable for this example	00000001	00	00	100101	10010100
	00000010	00	00	101111	10111100
	00000100	00	00	111011	11101100
	00001000	00	00	010011	01001100
	00010000	00	00	100110	10011000
	00100000	00	00	101001	10100100
	01000000	00	00	110111	11011100
	10000000	00	00	001011	00101100
	00000000	01	00	010110	01011001
	00000000	10	00	101100	10110010
	00000000	00	01	011001	01100100
	00000000	00	10	110010	11001000

O.6.3.4 Example of a “shifting zeroes” sequence

Table O.6 is not a SCSI sequence, but demonstrates the calculation results and theoretical output for twelve different codewords each containing all ones except for a single zero in a different bit in each word.

Table O.6 - Example of a “shifting zeroes” sequence

Information byte contents	Codeword bits (7:0) [DB(7:0)]	Codeword bits (9:8) [DB(9:8)]	Codeword bits (14:13) [Seq ID(1:0)]	Calculated redundant bits (5:0)	Theoretical output on DB(15:8)
Not applicable for this example	11111110	11	11	100101	10010111
	11111101	11	11	101111	10111111
	11111011	11	11	111011	11101111
	11110111	11	11	010011	01001111
	11101111	11	11	100110	10011011
	11011111	11	11	101001	10100111
	10111111	11	11	110111	11011111
	01111111	11	11	001011	00101111
	11111111	10	11	010110	01011010
	11111111	01	11	101100	10110001
	11111111	11	10	011001	01100111
	11111111	11	01	110110	11011011