Product Manual

SCSI Interface

Ultra 320

Ultra 160

Product Manual, Volume 2

Product Manual

SCSI Interface

Ultra 320

Ultra 160

Product Manual, Volume 2

Seagate

Publication number: 75789509, Rev. C
February 2004

Revision status summary sheet

Revision	Date	Sheets Affected	Sheets Affected
A Issue	01/31/2000	D. Ashby/G. Houlder	1/1, v thru viii, 1 thru 338.
Rev. B	04/22/2002	L. Newman/G. Houlder	1/1, v thru xvi, 1 thru 364.
Rev. C	02/13/2004	K. Schweiss/G. Holder	39, 83, 86, 94, 115, 123, 135-140, 141, 174, 202, 207, 210, 224-225, 227-228, 230, 232, 236, 238, 249, 252, 253, 255, 258, 260, 267, 280, 313, 325-326, 331-338, 345, 348, 351, 354-361, and 363.

Table of Contents

List of Figures

1.0 Interface requirements

1.1 How to use this interface manual

This manual provides a description of the SCSI[1] interface protocol and some general timing information as implemented by Seagate products. The features described in this manual are typically referred to as “Ultra160 SCSI” or “Ultra320 SCSI” features. Each individual drive’s Product Manual, Volume 1, for the various SCSI interface products contains additional and more detailed information on protocol, features supported, timing, and electrical/mechanical aspects of how the SCSI interface is implemented by that product.

This manual provides a general, tutorial-type description of the ANSI SCSI (formerly called SCSI-3) system. It is not intended to give all of the kinds of details needed to design/implement a SCSI system or product. For information about SCSI interface details not included herein or in Volume 1, refer to the standards listed in Section 1.1.1.

Note. The individual drive’s Product Manual, Volume 1, has tables that specify which SCSI features the drive implements, what the default parameters are for the various features they implement, which parameters are changeable, and which are not.

The combination of this specification together with the details in the individual drive’s Product Manual, Volume 1, provides a description of how a particular product implements the SCSI I/O system. This specification is Volume 2 of a set of manuals that is made up of an individual drive’s Product Manual, Volume 1, and this manual. The older Ultra2 SCSI Interface Product Manual, Volume 2, part number 77738479, applies to Seagate products that implement older versions of the SCSI interface (SCSI-1/SCSI-2). This new SCSI Interface Product Manual, Volume 2, part number 75789509, is referenced by newer individual drive’s Product Manuals, Volume 1, representing Seagate products that support Ultra160 or Ultra320 SCSI features and other new features, such as packetized information transfer (SPI information units), data group transfers, paced transfers, increased CRC protection, etc.

[1] Unless required for clarity, “SCSI” is now used instead of “SCSI-3.”

1.1.1 Scope of SCSI standards

Figure 1 uses a representative set of specifications to show the functional partitions and the relationships among SCSI standards applicable to drives covered by this product manual.

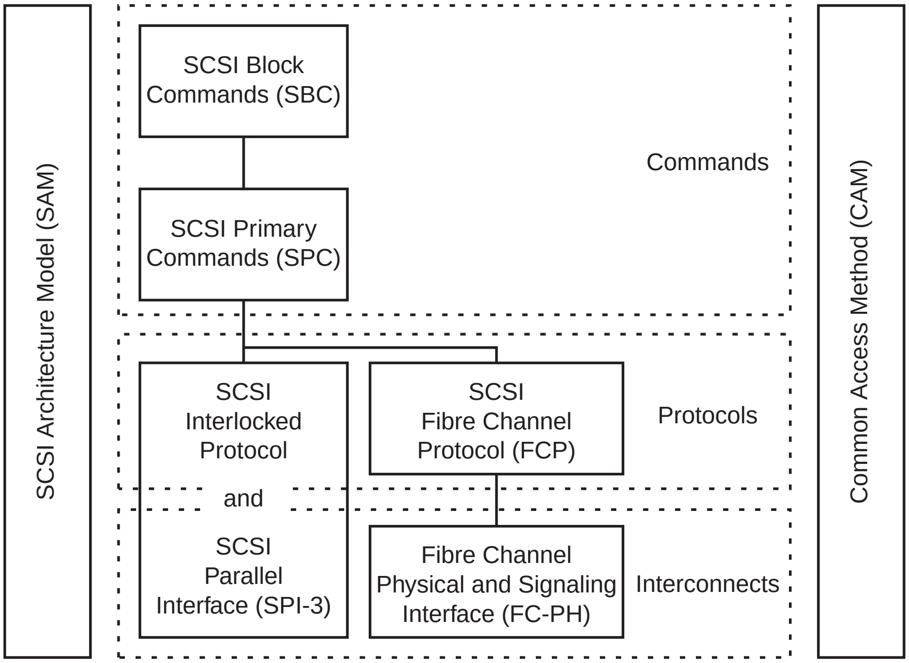

SCSI Architecture Model (SAM)
SCSI Block Commands (SBC)
SCSI Primary Commands (SPC)
Commands
SCSI Interlocked Protocol
and
SCSI Parallel Interface (SPI-3)
SCSI Fibre Channel Protocol (FCP)
Protocols
Fibre Channel Physical and Signaling Interface (FC-PH)
Interconnects
Common Access Method (CAM)

Figure 1. Functional scope of SCSI standards

The functional areas define the scope of each standard as follows:

- SCSI Architecture Model: Defines the SCSI systems model, the functional partitioning of the SCSI standard set and requirements applicable to all SCSI implementations and implementation standards.
- Commands: Implementation standards which define classes including a device model for each class. These standards specify the required commands and behavior that is common to all devices or unique to a given class of devices and prescribe the rules to be followed by a SCSI initiator port when sending commands to a device.
- Common Access Method: Implementation standard which defines a host architecture and set of services for device access.
- Protocols: Implementation standards which define the rules for exchanging information so that different SCSI devices can communicate.
- Interconnects: Implementation standards which define the electrical and signaling rules essential for devices to interoperate over a given physical interconnect.

The diagram of Figure 1 shows how the standards listed below fit within each category. The standards included in the diagram are meant to serve as examples and may not reflect the full set of standards currently in force.

1.1.2 Applicable standards

The following ANSI standards should be referenced for more details about SCSI system standards of operation:

- SCSI Architecture Model-2 (SAM-2), T10/1157D
- SCSI Enclosure Services Command Set (SES), T10/1212D
- SCSI Block Commands (SBC-2), T10/1417-D
- SCSI Primary Commands-3 (SPC-3), T10/1416-D
- SCSI Enhanced Parallel Interface (EPI), T10/1143D
- SCSI Parallel Interface (SPI-4), T10/1365D
- SCSI Medium Changer Commands (SMC), T10/999D
- SCSI Controller Command Set-2 (SCC-2), T10/1225D
- SCSI Stream Device Command Set (SSC), T10/997D
- SCSI Enclosure Services (SES), T10/1212D

1.2 General interface description

This SCSI Interface Product Manual describes the Seagate Technology, Inc. subset of the SCSI (Small Computer Systems Interface) as implemented on the Seagate-built drives. The interface is compatible with the SCSI Interface Specifications listed in Section 1.1.2. The drives covered by this product manual are classified as “Intelligent” peripherals.

The Seagate SCSI interface described herein consists of a 9 or 18 bit bidirectional data bus (includes bits for parity checking and enabling CRC protection), plus 9 control signals. The SCSI interface supports multiple initiators, disconnect/reconnect, self-configuring host software, automatic features that relieve the host from the necessity of knowing the physical architecture of the target (logical block addressing is used), and some other miscellaneous features.

The SCSI physical interface uses either single-ended drivers and receivers or low voltage differential drivers and receivers and uses asynchronous or synchronous communication protocols. The bus interface transfer rate for asynchronous or synchronous is given in individual drive’s Product Manual, Volume 1. The bus protocol supports multiple initiators, disconnect/reconnect, additional messages plus 6-byte, 10-byte, 12-byte, 16-byte and variable length Command Descriptor Blocks.

Unless specified otherwise in the individual drive's Product Manual, Volume 1, the drive is always a SCSI target port, and never a SCSI initiator port. For certain commands, which may or may not be supported by a particular drive model, the drive must act as a SCSI initiator port, but does not otherwise do so. For purposes of this specification, "drive" may be substituted for the word "target" wherever "target" appears.

In the event of a conflict between this document and ANSI SCSI documents, the requirements of the ANSI documents shall apply.

Note. In this revision, some new terminology is introduced as taken from the ANSI specifications. In many instances, the broader scope terms such as "initiator" and "target" are not used, but rather the more specific terms "initiator" and "target" appear. In Figure 2, it can be seen that several "initiators" from a single initiator may have one or more tasks in queue with several "targets" in a single target. A drive could be a SCSI target port or it could be one of the targets as part of some larger entity. When reading the description, one needs to be able to put the drive of interest in the proper context in terms of what is shown in Figure 2. For a proper understanding of the operation of the SCSI protocol, the terms in the SCSI architectural model as described in ANSI specification T10/1157D (SAM-2) should be well understood before reading operation descriptions in any SCSI document. Although a Glossary of terms is provided herein, the definitions may not be adequate for some. The SAM-2 specification gives a more detailed understanding of some of the new SCSI terminology

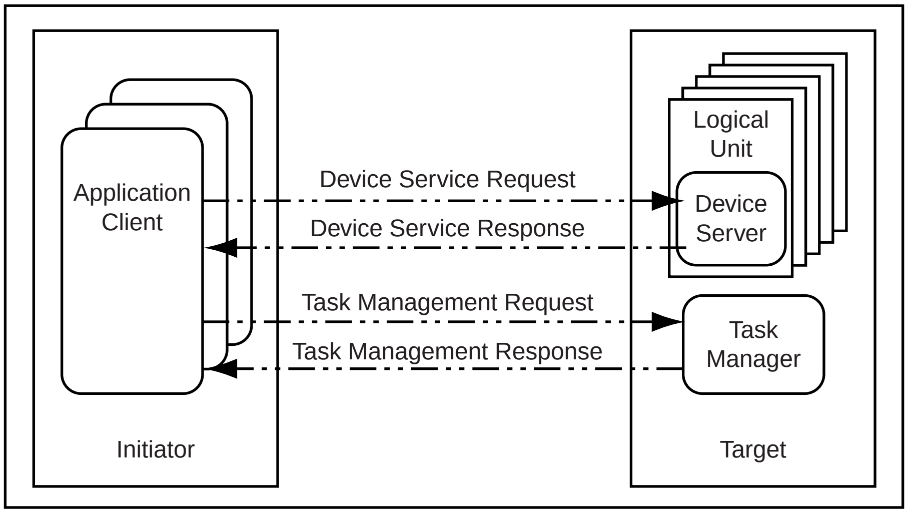


Figure 2. SCSI client-server model

1.2.1 Glossary

aborted command—A SCSI command that has been ended by aborting the task created to execute it.

ACA—Auto Contingent Allegiance (see below).

ACA command—A command performed by a task with the ACA attribute. See Section 4.4.2.

initiator—An object that is the source of SCSI commands. An object in this sense is not a tangible piece of hardware, but may be a single numeric parameter, such as a logical unit number, or a complex entity that performs a set of operations or services on behalf of another object (see ANSI SAM-2, T10/1157D).

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

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

auto contingent allegiance—One of the conditions of a task set following the return of a CHECK CONDITION status. See Section 4.4.2.

blocked (task state)—The state of a task that is prevented from completing due to an ACA condition.

blocking boundary—A task set boundary denoting a set of conditions that inhibit tasks outside the boundary from entering the Enabled state.

byte—An 8-bit construct.

call—The act of invoking a procedure.

client-server—A relationship established between a pair of distributed objects where one (the client) requests the other (the server) to perform some operation or unit of work on the client's behalf (see ANSI standard SAM-2, T10/1157D).

client—An object that requests a service from a server.

command—A request describing a unit of work to be performed by a target.

command descriptor block—A structure used to communicate a command from an initiator to a target. Command structures of 6, 10, 12, or 16 bytes are used, but a new variable length command structure has recently been introduced.

completed command—A command that has ended by returning a status and service response of Task Complete or Linked Command Complete.

completed task—A task that has ended by returning a status and service response of Task Complete. The actual events comprising the Task Complete response are protocol specific.

confirmation—A response returned to an object, which signals the completion of a service request.

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

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 Section 4.4.2.

control mode page—The mode page that identifies the settings of several target behaviors that may be of interest to an initiator or may be changed by an initiator. The complete definition of the Control mode page is found in Section 8.13.9.

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

cyclic redundancy check (CRC)—An error detecting code used to detect the validity of data that has been transferred during the current data group.

data field—The portion of a data group that contains data bytes.

data group—A sequence of data bytes and the four pCRC bytes during a DT DATA IN PHASE or a DT DATA OUT PHASE that starts at the first byte of the DT DATA phase or at the first byte after the last pCRC byte.

data group transfer—Parallel transfers that transfer data and pCRC information using only data groups. The last four bytes of a data group transfer contain CRC information over the whole data group.

destination device—The SCSI device to which a service delivery transaction is addressed. *See* source device.

target—An object within the logical unit which executes SCSI tasks according to the rules for task management described in clause 7 of ANSI SAM-2 document, T10/1157D.

device service request—A request, submitted by an initiator, conveying a SCSI command to a target.

device service response—The response returned to an initiator by a target on completion of a SCSI command.

differential—A signalling alternative that employs differential (two complementary signals) drivers and receivers to improve signal-to-noise ratios and increase maximum cable lengths.

disconnect—The action that occurs when a SCSI device releases control of the SCSI bus, allowing it to go to the BUS FREE PHASE.

domain—An I/O system consisting of a set of SCSI devices that interact with one another by means of a service delivery subsystem.

dormant (task state)—The state of a task that is prevented from starting execution due to the presence of certain other tasks in the task set.

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

driver—The circuitry used to control the state of the bus.

enabled (task state)—The state of a task that may complete at any time. Alternatively, the state of a task that is waiting to receive the next command in a series of linked commands.

ended command—A command that has completed or aborted.

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

faulted initiator—The initiator to which a Command Terminated or CHECK CONDITION status was returned.

faulted task set—A task set that contained a faulting task.

faulting command—A command that completed with a status of Check Condition or Command Terminated.

faulting task—A task that has completed with a status of Check Condition or Command Terminated.

function complete—A logical unit response indicating that a task management function has finished. The actual events comprising this response are protocol specific.

hard reset—a SCSI target port response to a reset event or a SCSI target port Reset in which the target performs the operations described in Section 7.6.7.

implementation—The physical realization of an object.

implementation-specific—A requirement or feature that is defined in a SCSI standard but whose implementation may be specified by the system integrator or vendor.

implementation option—An option whose actualization within an implementation is at the discretion of the implementor.

indication—The second step in a four-step confirmed service reply to a request.

information unit transfer—Parallel transfers that transfer data, status, commands, task attributes, task management information, acrid, and nexus information using only SPI information units.

initial connection—The result of a physical connect. It exists from the assertion of the BSY signal in a SELECTION PHASE until the next BUS FREE PHASE or the next QAS REQUEST message.

initiator—A SCSI device containing initiators which originate device service and task management requests to be processed by a SCSI target port SCSI device.

interconnect—The electrical media (including connectors and passive loads) used to connect the TERM-PWR, terminators, and SCSI devices in a SCSI bus.

interconnect subsystem—One or more physical interconnects which appear as a single path for the transfer of information between SCSI devices in a domain.

intersymbol interference (ISI)—The effect of adjacent symbols on the symbol currently being received.

in transit—Information that has been sent to a remote object but not yet received.

I/O operation—An operation defined by an unlinked SCSI command, a series of linked SCSI commands or a task management function.

I/O process—An I/O process consists of one initial connection or, if information units are enabled, the establishment of a nexus, and a zero or more physical or logical reconnection 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 ends with a SPI L_Q information unit with the type field set to status and the Data Length field set to zero.

I T nexus—A nexus that exists between a SCSI initiator port and a SCSI target port.

I T L nexus—A nexus that exists between a SCSI initiator port, a SCSI target port, and a logical unit. This relationship replaces the prior I T nexus.

I T L Q nexus—A nexus between a SCSI initiator port, a SCSI target port, a logical unit, and a queue tag following the successful receipt of one of the queue tag messages. This relationship replaces the prior I T L nexus.

iuCRC protection—The use of CRC to detect DT DATA PHASE data transmission errors during parallel transfers. Contains CRC information covering all bytes transmitted in a SPI information unit.

layer—A subdivision of the architecture constituted by subsystems of the same rank.

linked CDB—A CDB with the link bit in the control byte set to one.

linked command—One in a series of SCSI commands executed by a single task, which collectively make up a discrete I/O operation. In such a series, each command has the same task identifier, and all except the last have the link bit in the CDB control byte set to one.

logical connect—Establishes an I_T_L_Q nexus using SPI L_Q information units.

logical disconnect—Reduces the current I_T_L_Q nexus to an I_T nexus.

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

logical unit—a SCSI target port-resident entity which implements a device model and executes SCSI commands sent by an initiator.

logical unit number—A 64-bit identifier for a logical unit.

logical unit option—An option pertaining to a logical unit, whose actualization is at the discretion of the logical unit implementor.

lower level protocol—A protocol used to carry the information representing upper level protocol transactions.

mandatory—The referenced item is required to claim compliance with a standard.

media information—Information stored within a SCSI device which is non-volatile (retained through a power cycle) and accessible to a SCSI initiator port through the execution of SCSI commands.

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.

multimode single-ended (MSE)—A signalling alternative for multimode SCSI devices that employs MSE drivers and receivers to allow multimode SCSI devices to operate when SE SCSI devices are present on the bus.

nexus—A relationship between a SCSI initiator port and a SCSI target port, 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.

object—An architectural abstraction or “container” that encapsulates data types, services, or other objects that are related in some way.

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* parity bit. If an even number of asserted bits are detected at the receiver, a parity error occurs.

paced transfer—Parallel transfers that transfer information using pacing.

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.

packetized—A method of transferring information using SPI information units. *See* object.

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

parallel protocol request—Messages used to negotiate a synchronous data transfer agreement, a wide data transfer agreement, and set the protocol options between two SCSI devices.

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.

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

pCRC protection—The use of pCRC to detect DT DATA PHASE.

peer-to-peer protocol service—A service used by an upper level protocol implementation to exchange information with its peer.

peer entities—Entities within the same (protocol) layer.

pending task—A task that is not a current task.

physical interconnect—A single physical pathway for the transfer of information between SCSI devices in a domain.

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

physical reconnection—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 Section 3.1.1) or a QAS REQUEST message (see Section 4.3.13).

port—Synonymous with “service delivery port.” A single attachment to a SCSI bus from a SCSI device.

procedure—An operation that can be invoked through an external calling interface.

protocol—The rules governing the content and exchange of information passed between distributed objects through the service delivery subsystem.

protocol option—An option whose definition within a SCSI protocol standard is discretionary.

protocol service confirmation—A signal from the lower level protocol service layer notifying the upper layer that a protocol service request has completed.

protocol service indication—A signal from the lower level protocol service layer notifying the upper level that a protocol transaction has occurred.

protocol service request—A call to the lower level protocol service layer to begin a protocol service transaction.

protocol service response—A reply from the upper level protocol layer in response to a protocol service indication.

quick arbitration and selection process (QAS)—Quicker than the normal arbitration and selection process. Implementation is optional for SCSI devices.

queue—The arrangement of tasks within a task set, usually according to the temporal order in which they were created. *See* task set.

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

receiver—A client or server that is the recipient of a service delivery transaction.

reference model—A standard model used to specify system requirements in an implementation-independent manner.

request—A transaction invoking a service.

request-response transaction—An interaction between a pair of distributed, cooperating objects, consisting of a request for service submitted to an object followed by a response conveying the result.

request-confirmation transaction—An interaction between a pair of cooperating objects, consisting of a request for service submitted to an object followed by a response for the object confirming request completion.

reset event—A protocol-specific event which may trigger a hard reset response from a SCSI device as described in Section 5.3.

response—A transaction conveying the result of a request.

SCSI application layer—The protocols and procedures that implement or invoke SCSI commands and task management functions by using services provided by a SCSI protocol layer.

SCSI device—A device that contains at least one SCSI port and the means to connect its drivers and receivers to the bus.

SCSI device identifier—An address by which a SCSI device is referenced within a domain.

SCSI I/O system—An I/O system, consisting of two or more SCSI devices, a SCSI interconnect and a SCSI protocol, which collectively interact to perform SCSI I/O operations.

SCSI protocol layer—The protocol and services used by a SCSI application layer to transport data representing a SCSI application protocol transaction.

sender—A client or server that originates a service delivery transaction.

server—A SCSI object that performs a service on behalf of a client.

service—Any operation or function performed by a SCSI object, which can be invoked by other SCSI objects.

service delivery failure—Any non-recoverable error causing the corruption or loss of one or more service delivery transactions while in transit.

service delivery port—A device-resident interface used by the initiator, target or task manager to enter and retrieve requests and responses from the service delivery subsystem. Synonymous with “port.”

service delivery subsystem—That part of a SCSI I/O system which transmits service requests to a logical unit or target and returns logical unit or target responses to a SCSI initiator port.

service delivery transaction—A request or response sent through the service delivery subsystem.

signal—(n) A detectable asynchronous event possibly accompanied by descriptive data and parameters. (v) The act of generating such an event.

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

source device—The SCSI device from which a service delivery transaction originates. *See* destination device.

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

SPI L_Q information unit—The SPI L_Q information unit (see Section 6.2.2, tables 48 and 49) contains L_Q nexus (Logical unit—Q tag relationship) information for the information unit 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.

subsystem—An element in a hierarchically partitioned system which interacts directly only with elements in the next higher division or the next lower division of that system.

suspended information—Information stored within a logical unit that is not available to any pending tasks.

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

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.

task abort event—An event or condition indicating that the task has been aborted by means of a task management function.

task address—a SCSI initiator port identifies a task to a SCSI target port using a Task Address. The Task Address object represents either a Tagged Task Address or an Untagged Task Address without regard for the tagged or untagged nature of the Task. A Tagged Task Address is composed of a Logical Unit Identifier and a Tag. An Untagged Task Address is composed of a Logical Unit Identifier.

task completion event—An event or condition indicating that the task has ended with a service response of Task Complete.

task ended event—An event or condition indicating that the task has completed or aborted.

task management function—A task manager service which can be invoked by an initiator to affect the execution of one or more tasks.

t**ask management request**—A request submitted by an initiator, invoking a task management function to be executed by a task manager.

task management response—The response returned to an initiator by a task manager on completion of a task management request.

task manager—A server within the target which executes task management functions.

task set—A group of tasks within a SCSI target port device, whose interaction is dependent on the queuing and auto contingent allegiance rules of Section 7.6.1.

task slot—Resources within the logical unit that may be used to contain a task.

task tags—A Tag is a field containing up to 64 bits that is a component of a Tagged Task Identifier. A SCSI initiator port assigns tag values in each Tagged Task Identifier in a way that ensures that the identifier uniqueness requirements stated in ANSI SAM-2, T10/1157D, Section 4.9, are met.

third-party command—A SCSI command which requires a logical unit within the target device to assume the initiator role and send a SCSI command to a SCSI target port device.

transaction—A cooperative interaction between two objects, involving the exchange of information or the execution of some service by one object on behalf of the other.

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.

unconfirmed protocol service—A service available at the protocol service interface, which does not result in a completion confirmation.

unlinked command—A SCSI command having the link bit set to zero in the CDB control byte.

upper level protocol—An application-specific protocol executed through services provided by a lower level protocol.

1.2.2 Keywords

Several keywords are used to differentiate between different levels of requirements and optionality, as follows:

vendor-specific—Specification of the referenced item is determined by the device vendor.

protocol-specific—Implementation of the referenced item is defined by a SCSI protocol standard (see Section 1.1.1.)

expected—A keyword used to describe the behavior of the models specified by this standard.

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.

mandatory—A keyword indicating items required to be implemented as defined by this standard.

may—A keyword that indicates flexibility of choice with no implied preference (equivalent to “may or may not”).

may not—Keywords that indicates flexibility of choice with no implied preference (equivalent to “may or may not”).

obsolete—A keyword indicating items that were defined in prior SCSI standards but have been removed from this standard.

option, optional—Keywords that describe features which are not required to be implemented by this standard. However, if any optional feature defined by the standard is implemented, it shall be implemented as defined by the standard.

reserved—A key word 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 treated as an error.

shall—A keyword indicating a mandatory requirement. Designers are required to implement all such mandatory requirements to ensure interoperability with other standard conformant products.

should—A keyword indicating flexibility of choice with a strongly preferred alternative. Equivalent to the phrase “it is recommended.”

1.3 Physical interface characteristics

The physical interface characteristics (cables, connectors, electrical descriptions, termination requirements, etc.) for the drives covered by this Interface Manual are found in each individual drive's Product Manual, Volume 1, since these features are not the same for all drives.

1.4 Summary of SCSI commands and messages

Following is an alphabetical table listing the SCSI commands described in this manual. Details are given in Section 8.0.

Command name	Hex code	Device type	Page number
CHANGE DEFINITION	40	All	160
COMPARE	39	All	160
COPY	18	All	160
COPY AND VERIFY	3A	All	160
FORMAT UNIT	04	Direct access	161
INQUIRY	12	All	170
LOCK-UNLOCK CACHE (10)	36	Direct access	197
LOCK-UNLOCK CACHE (16)	92	Direct access	198
LOG SELECT	4C	All	200
LOG SENSE	4D	All	203
MODE SELECT (6)	15	Direct access	227
MODE SELECT (10)	55	Direct access	232
MODE SENSE (6)	1A	Direct access	235
MODE SENSE (10)	5A	Direct access	281
MOVE MEDIUM	A7		284
PERSISTENT RESERVE IN	5E	Direct access	285
PERSISTENT RESERVE OUT	5F	Direct access	292
PREFETCH (10)	34	Direct access	297
PREFETCH (16)	90	Direct access	299
PREVENT/ALLOW MEDIUM REMOVAL	1E	Direct access	300
READ (6)	08	Direct access	301
READ (10)	28	Direct access	303
READ (12)	A8	Direct access	305
READ (16)	88	Direct access	307
READ BUFFER	3C	All	309
READ CAPACITY (10)	25	Direct access	313
READ CAPACITY (16)	9E	Direct access	315
READ DEFECT DATA (10)	37	Direct access	317
READ DEFECT DATA (12)	B7	Direct access	319
READ ELEMENT STATUS	B4	Direct access	321
REASSIGN BLOCKS	07	Direct access	324
RECEIVE DIAGNOSTIC RESULTS	1C	All	326
RELEASE (6)	17	Direct access	330
RELEASE (10)	57	Direct access	330
REPORT DEVICE IDENTIFIER	A3	All	331
REPORT LUNS	A0	Direct access	333
REQUEST SENSE	03	All	336
RESERVE (6)	16	Direct access	348
RESERVE (10)	56	Direct access	348
REZERO UNIT	01	Direct access	348
SEARCH DATA EQUAL	31	Direct access	348
SEARCH DATA HIGH	30	Direct access	348
SEARCH DATA LOW	32	Direct access	348
SEEK	0B	Direct access	348
SEEK EXTENDED	2B	Direct access	349
SEND DIAGNOSTIC	1D	All	350
SET DEVICE IDENTIFIER	A4	All	354
SET LIMITS	33	Direct access	355
START/STOP UNIT	1B	Direct access	356
SYNCHRONIZE CACHE (10)	35	Direct access	357

Command name	Hex code	Device type	Page number
SYNCHRONIZE CACHE (16)	91	Direct access	358
TEST UNIT READY	00	All	360
VERIFY (10)	2F	Direct access	361
VERIFY (12)	AF	Direct access	363
VERIFY (16)	8F	Direct access	365
WRITE (6)	0A	Direct access	367
WRITE (10)	2A	Direct access	369
WRITE (12)	AA	Direct access	371
WRITE (16)	8A	Direct access	373
WRITE AND VERIFY (10)	2E	Direct access	375
WRITE AND VERIFY (12)	AE	Direct access	377
WRITE AND VERIFY (16)	8E	Direct access	379
WRITE BUFFER	3B	All	381
WRITE LONG	3F	Direct access	386
WRITE SAME (10)	41	Direct access	387
WRITE SAME (16)	93	Direct access	388
XD READ (10)	52	Direct access	390
XD READ (32)	7F	Direct access	391
XD WRITE (10)	50	Direct access	393
XD WRITE (32)	7F	Direct access	394
XD WRITEREAD (10)	53	Direct access	396
XD WRITEREAD (32)	7F	Direct access	397
XP WRITE (10)	51	Direct access	399
XP WRITE (32)	7F	Direct access	400

Following is an alphabetical summary of the SCSI messages described in this manual. Details are given in Section 4.

Message Name	Hex Code	Page number
ABORT TASK	06	110
CLEAR QUEUE	0E	111
DISCONNECT	04	91
EXTENDED MESSAGE	01	88
IDENTIFY	80-FF	91
IGNORE WIDE RESIDUE	23	92
INITIATE RECOVERY	0F	Not described in this manual
INITIATOR DETECTED ERROR	05	93
MESSAGE PARITY ERROR	09	93
MESSAGE REJECT	07	93
MODIFY DATA POINTER	01, 05, 00	94 (extended message)
NO OPERATION	08	95
PARALLEL PROTOCOL REQUEST	01, 06, 04	96
QAS REQUEST	55	100
RELEASE RECOVERY	10	Not described in this manual
RESTORE POINTERS	03	100
SAVE DATA POINTERS	02	100
SYNCHRONOUS DATA TRANSFER REQUEST	01, 03, 01	101 (extended message)
TASK ATTRIBUTE MESSAGES		107
ACA (AUTO CONTINGENT ALLEGIANCE)	24	108
HEAD OF QUEUE TAG	21	109
LINKED COMMAND COMPLETE	0A	93
ORDERED QUEUE TAG	22	109
SIMPLE QUEUE TAG	20	109
TASK COMPLETE	00	104
TASK MANAGEMENT MESSAGES		110
ABORT TASK	00	110
ABORT TASK SET	06	111
CLEAR ACA	16	111
CLEAR TASK SET	0E	111
LOGICAL UNIT RESET	17	111
TARGET RESET	0C	111
WIDE DATA TRANSFER REQUEST	01, 03	104 (extended message)

2.0 SCSI bus

This manual discusses only the “logical” and timing characteristics of the SCSI system and interface. A general description of the SCSI bus physical characteristics (voltages, connector configurations, pinouts, etc.) are given in the individual drive’s Product Manual, Volume 1, in the “Interface requirements” section, which covers the interface requirements and SCSI features supported by the drive described in the particular Product Manual being referenced. For all of the physical details of the SCSI interface, consult the ANSI standards referenced in Section 1.1.2.

Communication on the SCSI Bus is allowed between only two SCSI devices at a time. Seagate drives support systems with a maximum of 16 SCSI devices including the host computer(s) connected to the SCSI bus. Each SCSI device has a SCSI ID bit assigned as shown in Table 1. The SCSI ID is assigned by installing 0-3 jumper plugs onto a connector in a binary coded configuration during system configuration. Some drive models have an interface that includes the SCSI bus ID lines, so that the host can set the drive ID over the interface (see individual drive’s Product Manual, Volume 1, “Option/configuration headers” section).

When two SCSI devices communicate on the SCSI Bus, one acts as a SCSI initiator port and the other acts as a SCSI target port. The initiator (typically a host computer) originates an I/O process and the target performs whatever operations/tasks are called for by the I/O process. Devices arbitrate to get control of the bus to perform whatever communications are required by the current I/O process. As part of the arbitration process, devices on the SCSI bus assert their bus ID (one of the DB lines as shown in Table 1). The arbitration process is discussed in more detail later. Devices supported by this interface specification always operate as targets, unless otherwise specified (i.e., if certain commands are supported) in the individual drive’s Product Manual, Volume 1.

Table 1: SCSI IDs and associated SCSI bus arbitration priorities

SCSI ID/DB line asserted	SCSI bus arbitration priority
0 / DB0	#8
1 / DB1	#7
2 / DB2	#6
3 / DB3	#5
4 / DB4	#4
5 / DB5	#3
6 / DB6	#2
7 / DB7	#1 (highest)
8 / DB8	#16 (lowest)
9 / DB9	#15
10 / DB10	#14
11 / DB11	#13
12 / DB12	#12
13 / DB13	#11
14 / DB14	#10
15 / DB15	#9

The Host Adapter/Initiator must be identified by one of the 16 SCSI Device IDs (usually 7, which is highest arbitration priority). Make sure that none of the devices on the SCSI bus have duplicate IDs.

Certain SCSI bus functions are assigned to the initiator and certain SCSI bus functions are assigned to the target. The initiator will select a particular target. The target will request the transfer of Command, Data, Status, or other information on the data bus.

Under SCSI-2 protocol, information transfers on the data bus are interlocked and follow a defined REQ/ACK Handshake protocol. One byte of information will be transferred with each handshake. Synchronous data transfers do not require a one-for-one interlocking of REQ/ACK signals, but the total number of REQ pulses in a particular data transfer event must equal the total number of ACK pulses. The synchronous data transfer option is described in Paragraphs 3.1.5.2 and 3.5.3.2 of the Ultra2 SCSI Interface Product Manual, Volume 2, part number 77738479.

This Ultra160/Ultra320 SCSI Interface Product Manual, Volume 2, discusses SCSI-3 protocol (now called only “SCSI”). There are now different ways of latching data from the REQ and ACK signals, depending on whether ST DATA phases, DT DATA phases, or paced transfers are being used for information transfers. This is described in Section 2.1.2.

The drive supports single initiator, single target; single initiator, multiple target; multiple initiator, single target; or multiple initiator, multiple target bus configurations.

2.1 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. 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. The SCSI devices may be any combination of SCSI initiator ports (commonly called “initiators”) and SCSI target ports (commonly called “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 the initiator and the other acts as the target. The initiator originates an I/O process and the target receives the I/O process.

Some drive models have a single 80-pin I/O connector that contains additional interface lines that carry drive configuration select signals. These are peculiar to certain drives and are not SCSI standard signals. These are described in the individual drive’s product manual, Volume 1.

The 28 SCSI standard signals are described as follows:

BSY (Busy)—An “OR-tied” signal to indicate the bus is being used.

SEL (Select)—An “OR-tied” signal used by a SCSI initiator port to select a SCSI target port, or by a SCSI target port to reselect a SCSI initiator port.

RST (Reset)—An “OR-tied” signal that indicates the bus reset condition (see Section 5.2).

C/D (Control/Data)—A signal sourced by a SCSI target port that indicates whether CONTROL or DATA PHASE information is on the data bus. Assertion indicates Control (i.e., COMMAND, STATUS, and MESSAGE phases).

I/O (Input/Output)—A signal sourced by a SCSI target port to control the direction of data movement on the Data Bus with respect to a SCSI initiator port. Assertion indicates input to the initiator. This signal also distinguishes between SELECTION and RESELECTION phases.

MSG (Message)—A signal sourced by a SCSI target port during the MESSAGE PHASE.

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

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

ATN (Attention)—A signal sourced by a SCSI initiator port to indicate the Attention condition.

DIFFSENS (Differential Sense)/**Multimode—SE or LVD alternative**—“LW” and “LC” models have I/O circuits that can operate either in single-ended (SE) or low voltage differential (LVD) mode. When the interface DIFFSENS line is between 0 V and 0.6 V, the drive interface circuits operate single-ended. When DIFFSENS is between +0.7 V and +1.9 V, the drive interface circuits operate low voltage differential. This arrangement is not intended to allow dynamically changing transmission modes, but rather to prevent incompatible devices from attempting to interoperate. Drives must operate only in the mode for which the installation and interface cabling is designed. Multimode I/O circuits used by “LW” and “LC” devices do not operate at high voltage differential levels and should never be exposed to high voltage differential environments unless the command mode voltages in the environment are controlled to safe levels for single-ended and low voltage differential devices (see the ANSI SPI-4 specification). High Voltage Differential (HVD) is now an obsolete ANSI standard.

P_CRCA (Parity/CRC Available)—A signal identifying either parity or CRC available based on bus phase and negotiated settings.

During the SELECTION PHASE, RESELECTION PHASE, ST DATA PHASE, COMMAND PHASE, MESSAGE PHASE, and STATUS PHASE, this signal is referred to as DB(P_CRCA) and is sourced by the SCSI device port driving the Data Bus. The DB(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 DB(P_CRCA) bit is driven such that the number of logical ones in the byte plus the parity bit is odd.

Data group transfers are enabled (see Section 4.3.12) when this signal is referred to as P_CRCA and is sourced by the target to control whether a data group field is a pad field, pCRC field, or data field (see Section 2.11.1). When asserted, the data group field shall be pad or pCRC fields that shall not be transferred to the initiator. When negated, the data group field shall be a data field that shall be transferred to the initiator.

During DT DATA phases when information unit transfers are enabled, this signal is referred to as P_CRCA and sourced by the SCSI target. Depending on the negotiated condition of read streaming and write flow control, the SCSI initiator and target usage for P_CRCA is different. When information unit transfers are enabled, the SCSI target and initiator shall use the P_CRCA signal as indicated in Table 2.

Table 2: P_CRC signal usage requirements

Write flow control	Read streaming	DT Data phase	SCSI initiator response to P_CRCA	SCSI target usage of P_CRCA
Disabled	Disabled	All	Ignore	Continuously negated.
Enabled	Disabled	DT DATA IN	Ignore	Continuously negated.
		DT DATA OUT	Monitor	Asserts to indicate when the current SPI data stream information unit is the last SPI data stream information unit of the current write stream.
Disabled	Enabled	DT DATA IN	Monitor	Asserts to indicate when the current SPI data stream information unit is the last SPI data stream information unit of the current read stream.
		DT DATA OUT	Ignore	Continuously negated.
Enabled	Enabled	DT DATA IN	Monitor	Asserts to indicate when the current SPI data stream information unit is the last SPI data stream information unit of the current read stream.
		DT DATA OUT	Monitor	Asserts to indicate when 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 Section 4.3.12), 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 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.

DB(7–0) (8-bit data bus)—Each data bit that forms the 8-bit data bus. Bit significance and priority during arbitration are shown in Table 1.

DB(15–0) (16-bit data bus)—16 data bit signals that form the 16-bit Data Bus. Bit significance and priority during arbitration are shown in Table 1.

Greater detail on each of the SCSI bus signals is found in the following sections.

2.1.1 Drive select

For SCSI ID selection, install drive select jumpers as shown in configuration selection figure in the individual drive's Product Manual, Volume 1. Refer to the "Physical interface" section of the individual drive's Product Manual for the location of the drive select header. Drives using the 8-bit data interface can have one of eight ID bits selected by installing 0 through 2 (3) jumpers in a binary coded configuration on the drive select header. Drives using the 16-bit data interface can have one of 16 ID bits selected by installing 0 through 3 (4) jumpers in a binary coded configuration on the drive select header. "LC" model drives (80-pin direct connect I/O connector) can be assigned their bus ID over the SCSI interface.

2.1.2 Signal values

Signals may assume true or false values. There are two methods of driving these signals. In both cases, the signal shall be actively driven true, or asserted. In the case of OR-tied drivers, the driver does not drive the signal to the false state, rather the bias circuitry of the bus terminators pulls the signal false whenever it is released by the drivers at every SCSI device. If any driver is asserted, then the signal is true. In the case of non-OR-tied drivers, the signal may be negated. Negated means that the signal may be actively driven false, or may be simply released (in which case the bias circuitry pulls it false), at the option of the implementor.

2.2 Signal states

2.2.1 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. 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.

2.2.2 LVD signals

Figure 3 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.

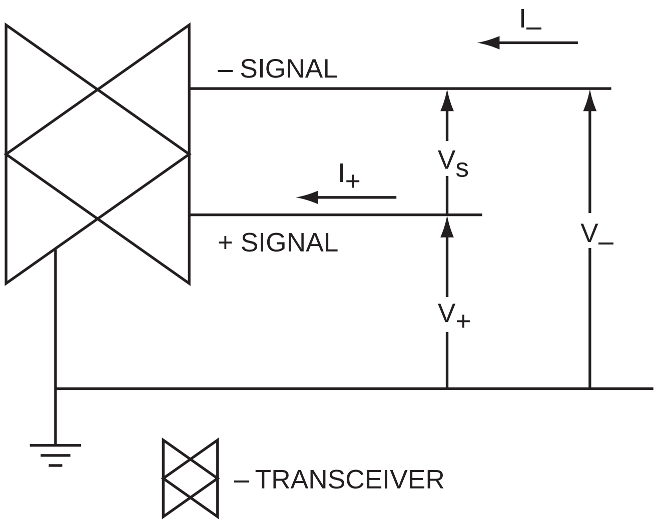


Figure 3. **Voltage and current definitions**

Figure 4 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.

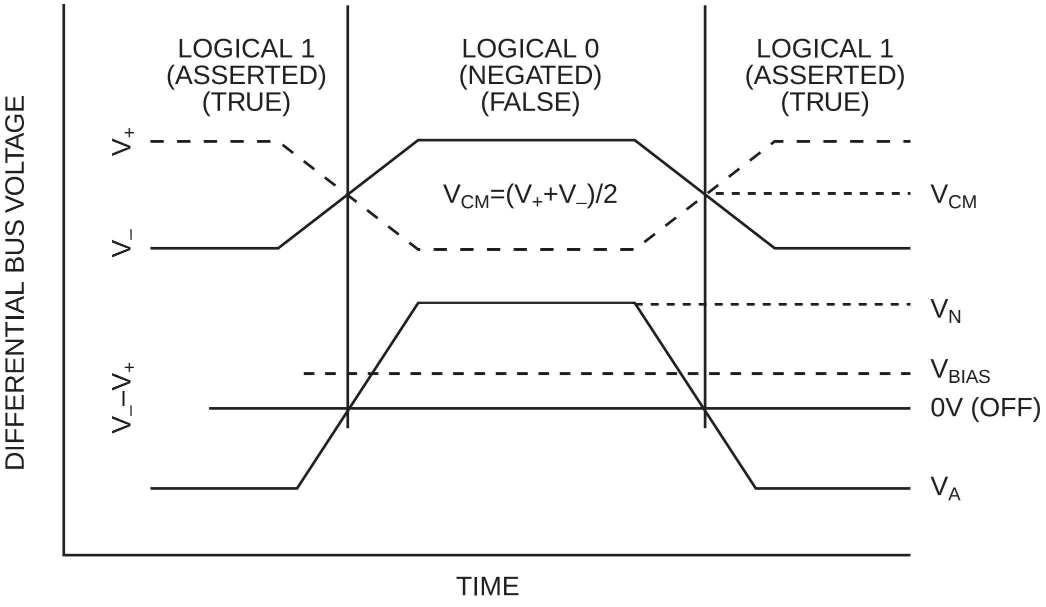


Figure 4. **LVD Signaling sense**

Note. For a description of V_{BIAS} see Section 7.3.1 of ANSI specification (SPI-4), T10/1365D.

2.3 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.

2.4 Signal sources

Table 3 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 3: Signal sources

	16 data bit (P) cable						
	8 data bit (A) cable						
SCSI bus phase	**BSY**	**SEL**	**C/D, I/O, MSG, REQ**	**ACK, ATN**	**DB(7-0)**	**P_CRCA**	**DB(15-8), DB(P1)**
BUS FREE	None	None	None	None	None	None	None
ARBITRATION (NORMAL)	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

Table abbreviations are defined as follows:

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, this 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.

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

PT: Target that initiated the QAS arbitration.

2.5 SCSI bus timing

Unless otherwise indicated, the delay time measurements for each SCSI device, defined in paragraphs 2.5.1 through 2.5.58 shall be calculated from signal conditions existing at that SCSI device's own SCSI bus connection. Thus, these measurements (except skew delay) can be made without considering delays in the cable. Refer to the tables below for the actual timing values for these delays.

Table 4: SCSI bus control timing values

Timing description	Reference	Type	Timing value
Arbitration delay	Section 2.5.1	minimum	2.4 µs
Bus clear delay	Section 2.5.4	maximum	800 ns
Bus free delay	Section 2.5.5	minimum	800 ns
Bus set delay	Section 2.5.6	maximum	1.6 µs
Bus settle delay	Section 2.5.7	minimum	400 ns
Cable skew [1]	Section 2.5.8	maximum	4 ns
Data release delay	Section 2.5.20	maximum	400 ns
DIFFSENS voltage filter time	Section 2.5.21	minimum	100 ms
Physical disconnection delay	Section 2.5.22	minimum	200 µs
Power on to selection [1]	Section 2.5.23	maximum	10 s
QAS arbitration delay	Section 2.5.24	minimum	1000 ns
QAS assertion delay	Section 2.5.25	maximum	200 ns
QAS release delay	Section 2.5.26	maximum	200 ns
QAS non-data phase REQ(ACK) period	Section 2.5.27	minimum	50 ns
Reset delay	Section 2.5.40	minimum	200 ns
Reset hold time	Section 2.5.41	minimum	25 µs
Reset to selection [2]	Section 2.5.42	maximum	250 ms
Selection abort time	Section 2.5.44	maximum	200 µs
Selection time-out delay [2]	Section 2.5.45	minimum	250 ms
System deskew delay	Section 2.5.49	minimum	45 ns

[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 5: SCSI bus data and information phase ST timing values

Timing description	Reference	Type	Timing values for negotiated transfer rate [1]				
			Asynch	Fast-5	Fast-10	Fast-20	Fast-40
ATN transmit setup time	Section 2.5.2	min	90 ns	33 ns	33 ns	21.5 ns	19.25 ns
ATN receive setup time	Section 2.5.3	min	45 ns	17 ns	17 ns	8.5 ns	6.75 ns
Cable skew [2]	Section 2.5.8	max	4 ns	4 ns	4 ns	3 ns	2.5 ns
Receive assertion period [3]	Section 2.5.28	min	N/A	70 ns	22 ns	11 ns	6.5 ns
Receive hold time [3] [4]	Section 2.5.29	min	N/A	25 ns	25 ns	11.5 ns	4.75 ns
Receive negation period [3]	Section 2.5.32	min	N/A	70 ns	22 ns	11 ns	6.5 ns
Receive setup time [3] [4]	Section 2.5.33	min	N/A	15 ns	15 ns	6.5 ns	4.75 ns
Receive REQ(ACK) period tolerance	Section 2.5.34	min	N/A	1.1 ns	1.1 ns	1.1 ns	1.1 ns
Signal timing skew	Section 2.5.46	max	8 ns	8 ns	8 ns	5 ns	4.5 ns
REQ(ACK) period	Section 2.5.39	nom	N/A	200 ns	100 ns	50 ns	25 ns
Transmit assertion period [2]	Section 2.5.53	min	N/A	80 ns	30 ns	15 ns	8 ns
Transmit hold time [3] [4]	Section 2.5.54	min	N/A	53 ns	33 ns	16.5 ns	9.25 ns
Transmit negation period [3]	Section 2.5.56	min	N/A	80 ns	30 ns	15 ns	8 ns
Transmit setup time [3] [4]	Section 2.5.57	min	N/A	23 ns	23 ns	11.5 ns	9.25 ns
Transmit REQ(ACK) period tolerance	Section 2.5.58	max	N/A	1 ns	1 ns	1 ns	1 ns

[1] 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.

[2] Cable skew is measured at each SCSI device connection within the same bus segment with the transmitted skew subtracted from the received skew.

[3] See Section 2.6 for measurement points for the timing specifications.

[4] See Section 9.6 in the ANSI SCSI Parallel Interface-4 (SPI-4) specification for examples of how to calculate setup and hold timing.

Table 6: Miscellaneous SCSI bus data and information phase DT timing values

Timing description	Reference	Type	Timing values for negotiated transfer rate [1]				
			Fast-10	Fast-20	Fast-40	Fast-80	Fast-160
Cable skew [2]	Section 2.5.8	max	4 ns	3 ns	2.5 ns	2.5 ns	2.5 ns
REQ(ACK) period	Section 2.5.39	nom	200 ns	100 ns	50 ns	25 ns	12.5 ns
Residual skew error [3]	Section 2.5.43	max	N/A	N/A	N/A	N/A	+/– 0.15 ns
De-skewed data valid window [4]	Section 2.5.11	min	N/A	N/A	N/A	N/A	+/– 2.1 ns
Skew correction range [4]	Section 2.5.47	min	N/A	N/A	N/A	N/A	+/– 3.65 ns [5]
Signal timing skew	Section 2.5.46	max	26.8 ns	13.4 ns	6.7 ns	3.35 ns	4.85 ns
Strobe offset tolerance	Section 2.5.48	max	N/A	N/A	N/A	N/A	+/– 0.125 ns

[1] 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.

[2] Cable skew is measured at each SCSI device connection within the same bus segment with the transmitted skew subtracted from the received skew.

[3] Calculated assuming timing budget shown in Table 9.

[4] Measured at the receiver terminal using clean input signals with 500 mV peak amplitude and 1 ns rise and fall time between 20% and 80% of the signal.

[5] Relative to the REQ(ACK) clocking signal.

Note. Fast-160 SCSI devices shall not change timing parameters between training (see Section 3.5.3.1) or reset events (see Section 5.4).

Table 7: Transmit SCSI bus data and information phase DT timing values

Timing description	Reference	Type	Timing values for negotiated transfer rate [1]				
			Fast-10	Fast-20	Fast-40	Fast-80	Fast-160
ATN transmit setup time	Section 2.5.2	min	48.4 ns	29.2 ns	19.6 ns	14.8 ns	14 ns
Flow control transmit hold time	Section 2.5.14	min	38.4 ns	19.2 ns	9.6 ns	4.8 ns	14 ns
Flow control transmit setup time	Section 2.5.15	min	48.4 ns	29.2 ns	19.6 ns	14.8 ns	14 ns
pCRC transmit hold time	Section 2.5.18	min	38.4 ns	19.2 ns	9.6 ns	4.8 ns	N/A
pCRC transmit setup time	Section 2.5.19	min	48.4 ns	29.2 ns	19.6 ns	14.8 ns	N/A
Transmit assertion period [2]	Section 2.5.53	min	92 ns	46 ns	23 ns	11.5 ns	5.69 ns
Transmit hold time [2] [3]	Section 2.5.54	min	38.4 ns	19.2 ns	9.6 ns	4.8 ns	4.77 ns
Transmit ISI compensation	Section 2.5.55	max					1.0 ns
Transmit negation period [2]	Section 2.5.56	min	92 ns	46 ns	23 ns	11.5 ns	5.69 ns
Transmit REQ(ACK) period tolerance	Section 2.5.58	max	0.6 ns	0.6 ns	0.6 ns	0.6 ns	0.06 ns
Transmit REQ assertion period with P_CRCA transitioning	Section 2.5.59	min	97.5 ns	54 ns	35.5 ns	24 ns	N/A
Transmit setup time [2] [3]	Section 2.5.57	min	38.4 ns	19.2 ns	9.6 ns	4.8 ns	1.48 ns
Transmitter skew	Section 2.5.61	max	N/A	N/A	N/A	N/A	+/– 0.75 ns
Transmitter time asymmetry	Section 2.5.62	max	N/A	N/A	N/A	N/A	+/– 0.25 ns

[1] 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.

[2] See Section 2.6 for measurement points for the timing specifications.

[3] See Section 9.6 in the ANSI SCSI Parallel Interface-4 (SPI-4) specification for examples of how to calculate setup and hold timing.

Note. Fast-160 SCSI devices shall not change timing parameters between training or reset events.

Table 8: Receive SCSI bus data and information phase DT timing values

Timing description	Reference	Type	Timing values for negotiated transfer rate [1]				
			Fast-10	Fast-20	Fast-40	Fast-80	Fast-160
ATN receive setup time	Section 2.5.3	min	13.6 ns	7.8 ns	4.9 ns	3.45 ns	3 ns
Flow control receive hold time	Section 2.5.12	min	11.6 ns	5.8 ns	2.9 ns	1.45 ns	3 ns
Flow control receive setup time	Section 2.5.13	min	18.6 ns	12.8 ns	9.9 ns	8.45 ns	3 ns
pCRC receive hold time	Section 2.5.16	min	11.6 ns	5.8 ns	2.9 ns	1.45 ns	N/A
pCRC receive setup time	Section 2.5.17	min	18.6 ns	12.8 ns	9.9 ns	8.45 ns	N/A
Receive assertion period [2]	Section 2.5.28	min	80 ns	40 ns	20 ns	8.5 ns	4.74 ns
Receive hold time [2] [3]	Section 2.5.29	min	11.6 ns	5.8 ns	2.9 ns	1.45 ns	0.08 ns
Receive negation period [2]	Section 2.5.32	min	80 ns	40 ns	20 ns	8.5 ns	4.74 ns
Receive setup time [2] [3]	Section 2.5.33	min	11.6 ns	5.8 ns	2.9 ns	1.45 ns	6.33 ns
Receive REQ(ACK) period tolerance	Section 2.5.34	min	0.7 ns	0.7 ns	0.7 ns	0.7 ns	0.06 ns
Receive REQ negation period with P_CRCA transitioning	Section 2.5.36	min	85.5 ns	48 ns	32.5 ns	21 ns	N/A
Receive skew compensation	Section 2.5.37	max	N/A	N/A	N/A	N/A	N/A
Receive internal hold time [4]	Section 2.5.30	min	N/A	N/A	N/A	N/A	0.345 ns
Receive internal setup time	Section 2.5.31	min	N/A	N/A	N/A	N/A	0.345 ns

[1] 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.

[2] See Section 2.6 for measurement points for the timing specifications.

[3] See Section 9.6 in the ANSI SCSI Parallel Interface-4 (SPI-4) specification for examples of how to calculate setup and hold timing.

[4] Calculated assuming timing budget shown in Table 9.

Note. Fast-160 SCSI devices shall not change timing parameters between training or reset events.

Table 9: SCSI Fast-160 timing budget template

Item	Fast-160	Comments
REQ(ACK) period	12.5 ns	From Table 7
Transfer period	6.25 ns	REQ(ACK) period / 2
Ideal Setup/Hold	3.125 ns	REQ(ACK) period / 4
Non-compensatable (Early to Late)		**Worst-case total of + and – time shift unless otherwise noted**
REQ(ACK) period tolerance / 2	0.06 ns	Tolerance of transmitter plus measurement error [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	Worst-case pattern
ISI of REQ(ACK)	0.0 ns	Assumed to be negligible in given chip
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
Assumed 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** [2]
Minimum compensated internal HOLD (int) data valid window	**0.345 ns**	**(Transfer period – post compensation error) / 2**

[1] Tolerance adjusted for half cycle (i.e., transfer period).

[2] Timing budgets in previous standards neglected asymmetry and detection ambiguity and lumps chip noise, clock jitter, crosstalk, noise, ISI and receiver amplitude skew into other terms (e.g., signal distortion skew) and/or ignores the effects.

2.5.1 Arbitration delay

The minimum time a SCSI device shall wait from asserting BSY for arbitration until the data bus can be examined to see if arbitration has been won (see Section 3.1.2). There is no maximum time.

2.5.2 ATN transmit setup time

When information unit transfers are not being used, the ATN transmit setup time is the minimum time provided by the transmitter between the assertion of the ATN signal and the last negation of the ACK signal in any phase.

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

When information unit transfers are being used with paced transfers, the ATN transmit setup time is the minimum time provided by the transmitter between the assertion of the ATN signal and the assertion 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.

2.5.3 ATN receive setup time

When information unit transfers are not being used, the ATN receive setup time is the minimum time required at the receiver between the assertion of the ATN signal and the last negation of the ACK signal in any phase to recognize the assertion of an attention condition.

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

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

2.5.4 Bus clear delay

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

1. The BUS FREE phase is detected (i.e., the BSY and SEL signals are both false for a bus settle delay).
2. The SEL signal is received from another SCSI device during the ARBITRATION phase.
3. The transition of the RST signal to true.

For item 1 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.

2.5.5 Bus free delay

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

2.5.6 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.

2.5.7 Bus settle delay

The minimum time to wait for the bus to settle after changing certain control signals as specified in the protocol definitions. Provides time for a signal transition to propagate from the driver to the terminator and back to the driver.

2.5.8 Cable skew delay

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

2.5.9 Chip noise in receiver

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

2.5.10 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.

2.5.11 De-skewed data valid window

The minimum difference in time allowed between the rising or falling edge of a "1010..." pattern on the DAT BUS or DB(P1) and its clocking signal on the ACK or REQ signal as measured at their zero-crossing points after skew compensation is applied by the receiver without allowing any error in the received data. The de-skewed data valid window shall be equal to:

+/– [(data transfer period) – (residual skew error) – (strobe offset tolerance) – (clock jitter) – (receiver amplitude skew) – (ship noise) – (system noise at receiver) – (receiver asymmetry)] / 2.

2.5.12 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.

2.5.13 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.

2.5.14 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 stream 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.

2.5.15 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.

2.5.16 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 during data group transfers.

2.5.17 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 during data group transfers. Specified to ease receiver timing requirements and ensure that this signal, that is outside CRC protection, is received correctly.

2.5.18 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 during data group transfers.

2.5.19 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 during data group transfers. Specified to provide the increased receive setup time, subject to intersymbol interference, cable skew, and other distortions.

2.5.20 Data release delay

The maximum time for a SCSI initiator port 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.

2.5.21 DIFFSENS voltage filter time

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

2.5.22 Physical disconnection delay

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

2.5.23 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 ANSI SCSI Primary Commands-3 standard).

2.5.24 QAS arbitration delay

The minimum time a SCSI device with QAS enabled 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 Section 4.3.12).

2.5.25 QAS assertion delay

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

2.5.26 QAS release delay

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

2.5.27 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 the data is valid during the COMMAND, MESSAGE, and STATUS phases.

2.5.28 Receive assertion period

The minimum time required at a SCSI device receiving a REQ signal for the signal to be asserted while using synchronous transfers or paced transfers, provided P_CRCA is not transitioning. Also, the minimum time required at a SCSI device receiving an ACK signal for the signal to be asserted while using synchronous transfers or paced 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 ANSI specification, T10/1302D, for signal measurement points.

2.5.29 Receive hold time

For ST data transfers, the minimum time required 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 data transfers, provided P_CRCA is not transitioning during data group transfers. 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.

2.5.30 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 worst-case bit (i.e., data or parity) to the compensated offset strobe.

2.5.31 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 worst-case bit (i.e., data or parity) to the compensated offset strobe.

2.5.32 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 or paced transfers. Also, the minimum time required at a SCSI device receiving an ACK signal for the signal to be asserted while using synchronous transfers or paced 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 ANSI specification, T10/1302D, for signal measurement points.

2.5.33 Receive setup time

For ST 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 assertion of the REQ signal or the ACK signal while using synchronous data 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.

2.5.34 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.

2.5.35 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 P_CRCA is transitioning during data group transfers. Specified to ensure that the assertion period is longer than the receive hold time plus the receive setup time.

2.5.36 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 P_CRCA is transitioning during data group transfers. Specified to ensure that the negation period is longer than the receive hold time plus the receive setup time.

2.5.37 Receive skew compensation

The effective reduction in worst-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.

2.5.38 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.

2.5.39 REQ(ACK) period

The REQ(ACK) period during synchronous transfers or paced transfers is the nominal time between adjacent assertion edges of the REQ or ACK signal for the fastest negotiated 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 crosstalk and ISI the measurements should be made by averaging the time between edges during long (i.e., greater than 512 bytes) all zeros 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 that 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.

2.5.40 Reset delay

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

2.5.41 Reset hold time

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

2.5.42 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-3 Standard).

2.5.43 Residual skew error

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

2.5.44 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 timeout is required to ensure that a SCSI target port or initiator does not assert the BSY signal after a SELECTION or RESELECTION phase has been aborted.

2.5.45 Selection timeout delay

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

2.5.46 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 ANSI standard SPI-4, T10/1365D. The receiver detection range is the part of the signal between the “may detect” level and the “shall detect” level on either edge (see Section 9.3).

2.5.47 Skew correction range

The minimum skew correction capability of the receiver of a signal on the DATA BUS or DB(P1) relative to the ACK or REQ signal as measured at the receiver’s connector. The skew correction range shall be equal to:

+/– [(transmitter chip skew) + (cable skew) + (two times trace skew)]

relative to the corresponding ACK or REQ clock signal for that transition. Receiver chip skew is not included, as it is internal to the receiver.

2.5.48 Strobe offset tolerance

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

2.5.49 System deskew delay

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

2.5.50 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.

2.5.51 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.

2.5.52 Time asymmetry

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

2.5.53 Transmit assertion period

The minimum time that a SCSI target port shall assert the REQ signal while using synchronous transfers or paced transfers, provided it is not transitioning P_CRCA during data group transfers. Also, the minimum time that a SCSI initiator port shall assert the ACK signal while using synchronous transfers or paced transfers.

2.5.54 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) while using synchronous data 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).

2.5.55 Transmit ISI compensation

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

2.5.56 Transmit negation period

The minimum time that a SCSI target port shall negate the REQ signal while using synchronous transfers or paced transfers, provided it is not transitioning P_CRCA during data group transfers. Also, the minimum time that a SCSI initiator port shall negate the ACK signal while using synchronous transfers or paced transfers.

2.5.57 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) and the assertion of the REQ signal or the ACK signal while using synchronous data 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) and the transition of the REQ signal or the ACK signal.

2.5.58 Transmit REQ(ACK) period tolerance

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

2.5.59 Transmit REQ assertion period with P_CRCA transitioning

The minimum time that a SCSI target port shall assert the REQ signal while transitioning P_CRCA during data group transfers. Specified to provide the increased receive REQ assertion period, subject to loss on the interconnect.

2.5.60 Transmit REQ negation period with P_CRCA transitioning

The minimum time that a SCSI target port shall negate the REQ signal while transitioning P_CRCA during data group transfers. Specified to provide the increased receive REQ negation period, subject to loss on the interconnect.

2.5.61 Transmitter skew

The maximum difference in time allowed between the rising or falling edge of a “1010...” pattern on the DATA BUS or DB(P1) signal and its clocking signal on the ACK or REQ signal as measured at their zero-crossing points. The signals for the output waveforms shall be measured at the connector of the transmitting device.

2.5.62 Transmitter time asymmetry

The maximum time on DATA BUS, DB(P1), ACK, or REQ signal from any transition edge to the subsequent transition edge during a “1010...” pattern, as measured at their zero-crossing points, minus the data transfer period. The signals for the output waveforms shall be measured at the connector of the transmitting device.

2.6 Measurement points

The measurement points for single-ended (SE) and low voltage differential (LVD) ACK, REQ, DATA, P_CRCA, and PARITY signals are defined in ANSI standard SPI-4.

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

2.6.1 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 Section 9 of ANSI standard SPI-4, 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.

2.6.2 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 Section 9 of ANSI standard SPI-4, 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.

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

SE SCSI devices are not capable of Fast-40 and Fast-80 data transfer rates. LVD circuits are required for these faster transfer rates.

2.6.3 LVD measurement points

When transferring data using ST LVD phases, LVD SCSI devices shall use the measurement points defined in Section 9 of ANSI standard SPI-4, for the measurement of timing values. When transferring data using DT DATA phases, LVD SCSI devices shall use the measurement points defined in Section 9 of ANSI standard SPI-4, for the measurement of timing values. The rise and fall times for the LVD REQ/ACK signals shall be nominally the same as for the LVD Data, P_CRCA, and DP-1 signals.

2.7 Clocking methods for data transfers

This manual 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 at which the 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. An example of a negotiated transfer period of 25 ns with DT transfers is shown in Figure 7. For ST and DT synchronous transfers the clocking signal (i.e., REQ or ACK) occurs when the DATA BUS is in a steady state.

Figure 8 shows an example of transfers with a negotiated transfer period of 6.25 ns at the receiving SCSI device’s connector. 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. For paced transfers, the clocking signal (i.e., REQ or ACK) may occur when the data bus is changing state.

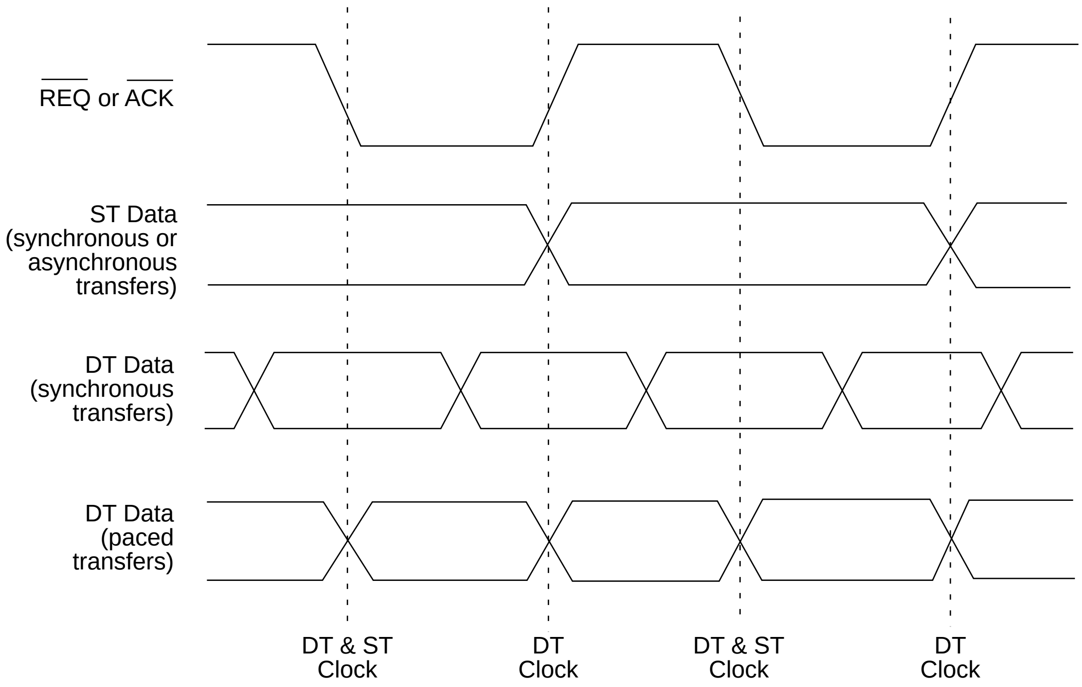
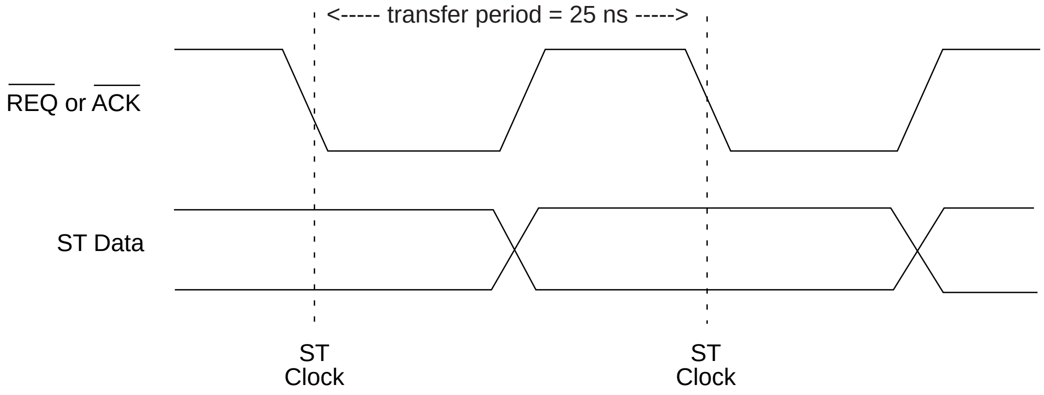
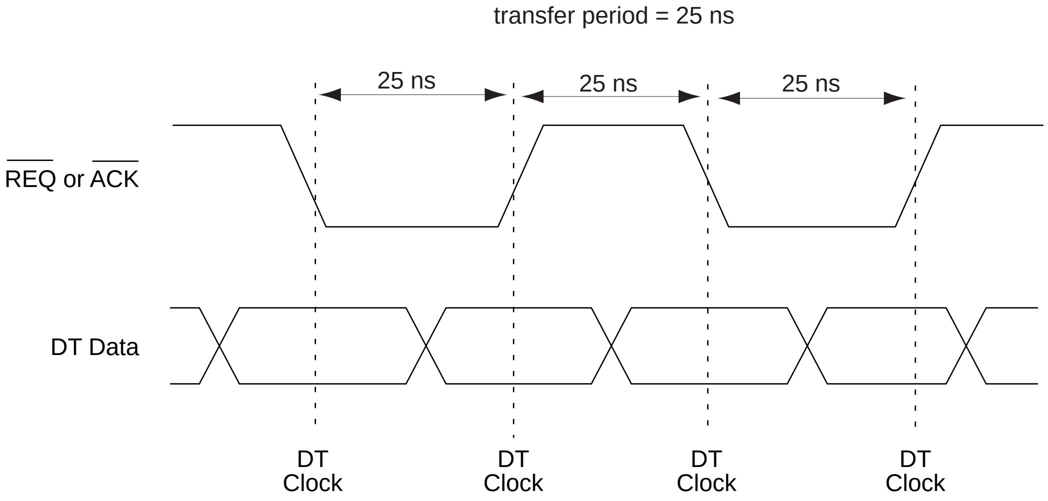

$\overline{\text{REQ}}$ or $\overline{\text{ACK}}$
ST Data (synchronous or asynchronous transfers)
DT Data (synchronous transfers)
DT Data (paced transfers)
DT & ST Clock
DT Clock
DT & ST Clock
DT Clock

Figure 5. ST latching data vs. DT latching data


<----- transfer period = 25 ns ----->
$\overline{\text{REQ}}$ or $\overline{\text{ACK}}$
ST Data
ST Clock
ST Clock

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

Figure 6. ST synchronous transfer example


transfer period = 25 ns
25 ns
25 ns
25 ns
$\overline{\text{REQ}}$ or $\overline{\text{ACK}}$
DT Data
DT Clock
DT Clock
DT Clock
DT Clock

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

Figure 7. DT synchronous transfer example

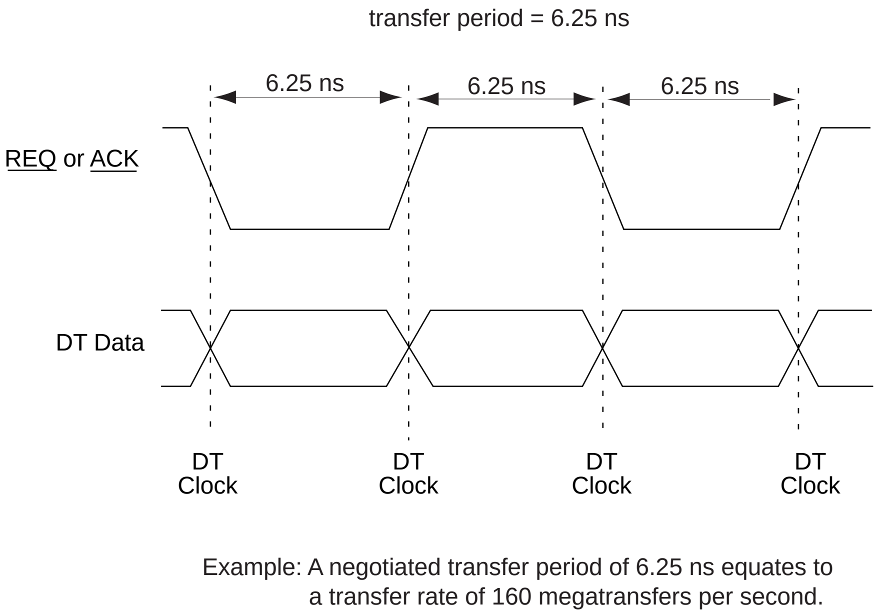

transfer period = 6.25 ns
6.25 ns
6.25 ns
6.25 ns
REQ or ACK
DT Data
DT Clock
DT Clock
DT Clock
DT Clock
Example: A negotiated transfer period of 6.25 ns equates to a transfer rate of 160 megatransfers per second.

Figure 8. Paced transfer example

2.8 Paced transfer on a SCSI bus

A SCSI bus that supports paced transfers has additional driver and receiver functions required over those used with synchronous transfers or asynchronous transfers. These functions include driver precompensation, receiver skew compensation, receiver clock shifting, and an optional receiver signal adjustment. In addition, the drive precompensation may be switched out of the data path at the request of the receiving SCSI device.

The receiver skew compensation and clock shifting adjust the timing relationship between the clocking signal (i.e., REQ or ACK) and the signals being clocked (e.g., the data bus signals). That adjustment causes the clock signal to align with the middle of the signals being clocked when those signals enter the receiver. The receiver is then able to the clock signal to latch valid data.

During paced transfers, the clock signal (i.e., REQ or ACK) transitions at the negotiated transfer period. Data is qualified by the clock signal and the phase of the P1 signal.

Receiver skew compensation is vendor specific and, therefore, not defined in this manual.

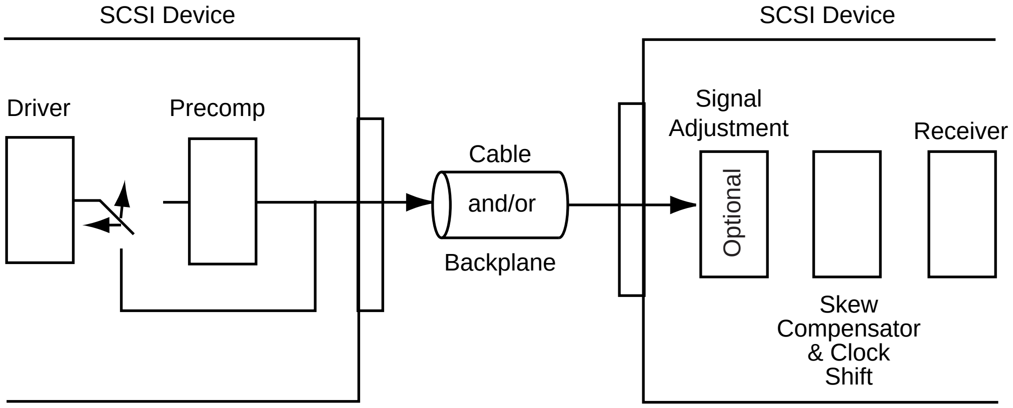


Figure 9. Example of a SCSI bus with paced transfers

2.9 Data transfer modes

There are three types of transfer modes:

- Asynchronous
- Synchronous
- Paced

This section provides a brief description of each of these types of transfer modes.

2.9.1 Asynchronous transfers

SCSI device ports default to 8-bit asynchronous transfers.

8-bit asynchronous transfers are used for all COMMAND, STATUS, and MESSAGE phases.

ST DATA phases may use 8-bit or 16-bit asynchronous transfers. Asynchronous transfers are not permitted in DT DATA phases.

2.9.2 Synchronous transfers

ST DATA phases shall use synchronous transfers when a synchronous transfer agreement is in effect. ST DATA phases may use 8-bit or 16-bit synchronous transfers.

DT DATA phases shall use synchronous transfers when a synchronous transfer agreement is in effect. DT DATA phases shall only use wide transfers.

2.9.3 Paced transfers

Paced transfers shall only be used in DT DATA phases when a paced transfer agreement is in effect. DT phases shall only use wide transfers.

2.10 ST DATA phase parallel transfers

The format of data transmitted during ST DATA phases consists of data and protection. Parity generation and checking give some error detection protection in the ST phase data.

2.11 DT DATA phase parallel transfers

During DT DATA phases, communicating SCSI devices format information according to one of two protocol options:

- Data group transfers. Data groups encapsulate all data and associated error protection.
- Information unit transfers. Information units encapsulate all nexus, task management, task attribute, command, data, and error protection. Usually these are called “SPI information units.”

Sections 2.11.1 and 2.11.2 contain a brief description of how packetized information transfer and CRC protection fit into the SCSI I/O system operation.

2.11.1 Data group transfers

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, 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 (see Section 4.3.4) with the Number of Bytes to Ignore field set to 01h.

During DT DATA OUT phase, if a SCSI target port 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.

2.11.2 Information unit transfers

Information unit transfers are permitted when a synchronous transfer agreement is in effect. Information unit transfers are mandatory when a paced transfer agreement is in effect. Information unit transfers are not permitted when an asynchronous transfer agreement is in effect.

During information unit transfers, each DT DATA IN phase and DT DATA OUT phases 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 as is necessary to make the transfer a multiple of four 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 bytes and iuCRC fields.

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. If specified, an iuCRC interval shall begin on the first transfer of each data information unit or data stream information unit.

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 the iuCRC is transmitted.

SPI data stream information units may be used to transfer data to or from a SCSI device. Support of data streaming during DT DATA OUT phases, called write streaming, is mandatory. Support of data streaming during DT DATA IN phases, called read streaming, is optional. The use of read streaming is part of the negotiated transfer agreement between two SCSI devices (i.e., the RD_STRM bit set to one). A SCSI target port is not required to use read streaming even if streaming support is enabled.

A SCSI target port, while streaming data, may give an indication that the stream of SPI data stream information nits are about to end while still sending the current SPI data stream information unit. This early warning is called flow control. Support of flow control during DT DATA OUT phases, called write flow control, is optional.

Support of flow control during DT DATA IN phases, called read flow control, is mandatory if read streaming is enabled. The use of write flow control is part of the negotiated transfer agreement between two SCSI devices (i.e., the WR_FLOW bit set to one).

2.12 Negotiation

PARALLEL PROTOCOL REQUEST (PPR) (see Section 4.3.12), SYNCHRONOUS DATA TRANSFER REQUEST (SDTR) (see Section 4.3.16), and WIDE DATA TRANSFER REQUEST (WDTR) (see Section 4.3.18) messages are used to alter the transfer agreement between two ports. The transfer agreement defines the protocol used during data phases (e.g., transfer period, REQ/ACK offset, transfer width) and agreement on features not affecting data phases (e.g., QAS). All other information transfer phases (i.e., COMMAND, MESSAGE, and STATUS) use eight-bit asynchronous data transfers.

PPR, SDTR, and WDTR messages are called negotiation messages. When a SCSI initiator port sends one of them, the message names are PPR OUT, SDTR OUT, and WDTR OUT. When a SCSI target port sends one of them, the message names are PPR IN, SDTR IN, and WDTR IN. A negotiation sequence consists of at least one matching set of negotiation messages (e.g., PPR OUT and PPR IN).

A transfer agreement is maintained by each port for each other port on the SCSI bus. Each port may be used as either a SCSI target port or a SCSI initiator port. The same transfer agreement applies whether the port is being used as a SCSI target port or as a SCSI initiator port.

2.12.1 Negotiation algorithm

A SCSI initiator port and SCSI target port exchange negotiation messages to perform negotiation. The originating port is the one that sends the first negotiation message and the responding port is the one that replies. Ports shall not set message fields to values they do not support. The originating port should set the fields in the originating negotiation message to the maximum values (e.g., fastest transfer period, largest REQ/ACK offset) it supports. If the responding port is able to support the requested values, it shall return the same values in the responding negotiation message. If the responding port requires different values (i.e., a subset of the originating port's request), it shall return those values in the responding negotiation message (e.g., if the originating port asks for a REQ/ACK offset of 32 and the responding port only supports a REQ/ACK offset of 16, then the responding port replies with an offset of 16).

If the responding negotiation message contains values the originating port does not support, the originating port shall respond with a MESSAGE REJECT message.

2.12.2 When to negotiate

Each port shall maintain a negotiation required flag for each other port. A port shall set its negotiation required flags to true for all other ports after a reset event. A port shall set its negotiation required flag to true for a given port after an error occurs while transmitting a responding negotiation message to that port.

A SCSI initiator port shall set its negotiation required flag to true for a SCSI target port after an unexpected COMMAND phase occurs when selecting without using attention condition (i.e., when selecting a SCSI target port with information units enabled).

A logical unit reset has no effect on negotiation required flags or on transfer agreements.

After a reset event a port shall set its transfer agreements for all other ports to the default transfer agreement (see Table 12).

A SCSI initiator port shall originate negotiation before sending a command to a SCSI target port whenever its negotiation required flag is true for that SCSI target port. A SCSI target port shall originate negotiation before accepting a command from a SCSI initiator port whenever its negotiation required flag is true for that SCSI initiator port. After successful negotiation or reaching the default transfer agreement, the negotiation required flag shall be set to false.

A port may originate negotiation even if its negotiation required flag is false (e.g., to change the settings, as part of integrity checking procedures, or, for a SCSI initiator port, after a SCSI target port has originated negotiation). Negotiation should not be originated after every selection and reselection as this may impact performance.

Note. SCSI target ports may have had their support for originating negotiation after power on disabled to support illegal SCSI initiator device software. If a SCSI initiator port sends a command to a SCSI target device that has been powered on (e.g., after a hot plug) that results in a unit attention condition, the SCSI initiator port determines that negotiation is required and originates negotiation before the next command. However, if the command is INQUIRY, REPORT LUNS, or REQUEST SENSE, a unit attention condition is not created. An invalid data phase may occur if the SCSI target port does not originate negotiation. If the SCSI initiator port always originates negotiation before sending those commands, the data phase runs correctly. When information units are disabled, a SCSI initiator port may originate negotiation with its currently negotiated settings before each INQUIRY, REPORT LUNS, or REQUEST SENSE command to avoid this problem. When information units are enabled, the selection without attention results in an unexpected COMMAND phase that notifies the SCSI initiator port that negotiation before each INQUIRY, REPORT LUNS, or REQUEST SENSE command is not needed.

2.12.3 Negotiable fields

Table 10 lists the fields that may be negotiated and the effects of successful negotiation on those fields by each of the different negotiation messages. Ports shall implement a given message if they implement fields that are negotiable with that message.

Table 10: Negotiable fields and effects of successful negotiation

Field name		Negotiation message pair		
		PPR	**WDTR**	**SDTR**
Transfer Period Factor		Negotiated (valid values: 08h-FFh)	No requirement	Negotiated (valid values: 0Ah-FFh)
REQ/ACK Offset		Negotiated	Sets to 00h	Negotiated
Transfer Width Exponent		Negotiated (valid values: 00h-01h)	Negotiated (valid values: 00h-01h)	Unchanged
Protocol options	PCOMP_EN	Negotiated	Sets to 00h	Sets to 00h
	RTI	Negotiated	Sets to 00h	Sets to 00h
	RD_STRM	Negotiated	Sets to 00h	Sets to 00h
	WR_FLOW	Negotiated	Sets to 00h	Sets to 00h
	HOLD_MCS	Negotiated	Sets to 00h	Sets to 00h
	QAS_REQ	Negotiated	Sets to 00h	Sets to 00h
	DT_REQ	Negotiated	Sets to 00h	Sets to 00h
	IU_REQ	Negotiated	Sets to 00h	Sets to 00h

When negotiating, the responding port shall respond with values that are a subset of the values in the originating message as indicated in Table 11 (e.g., if the originating message requests a REQ/ACK offset of 10h, the responding message has a REQ/ACK offset field set to 10h or lower).

Table 11: Responding message requirements

Field Name		Message	Response shall be numerically
Transfer Period Factor		PPR, SDTR	Greater than or equal
REQ/ACK Offset		PPR, SDTR	Less than or equal
Transfer Width Exponent		PPR, WDTR	00h or 01h
Protocol Options	PCOMP_EN	PPR	Less than or equal
	RTI	PPR	Less than or equal
	RD_STRM	PPR	Less than or equal
	WR_FLOW	PPR	Less than or equal
	HOLD_MCS	PPR	Less than or equal
	QAS_REQ	PPR	Less than or equal
	DT_REQ	PPR	Less than or equal
	IU_REQ	PPR	Less than or equal

2.12.4 Transfer agreements

The transfer agreements that are in effect for various combinations of field values are described in Table 12.

Table 12: Transfer agreements

Transfer agreement	REQ/ACK offset	Transfer period factor	Transfer width exponent	DT_REQ	IU_REQ	QAS_REQ	All other protocol options
Default	00h	Any	00h	0	0	0	0
Asynchronous	00h	Any	Any	0	0	0	0
ST synchronous	GE 01h	GE 0Ah	Any	0	0	0	0
DT synchronous	GE 01h	GE 09h	01h	1	Any	Any	Any
Paced	GE 01h	08h	01h	1	1	Any	Any
Wide	Any	Any	01h	Any	Any	Any	Any
Narrow	GE 0Ah	Any	00h	0	0	0	0
Data group	GE 09h	Any	01h	1	0	0	0
Information unit	GE 01h	08h	01h	1	1	Any	Any

2.12.5 Transfer period factor

The Transfer Period Factor field selects the transfer period and determines which transfer rate's timing values in Tables 5, 6, 7, and 8 shall be honored, provided that REQ/ACK OFFSET is greater than 00h. The field values are defined in Table 13.

Table 13: Transfer period factor

Value	Description	Message	Transfer rate
00h - 07h	Reserved	N/A	N/A
08h	Transfer period equals 6.25 ns	PPR	Fast-160
09h	Transfer period equals 12.5 ns	PPR	Fast-80
0Ah	Transfer period equals 25 ns	PPR, SDTR	Fast-40
0Bh	Transfer period equals 30.3 ns	PPR, SDTR	Fast-40
0Ch	Transfer period equals 50 ns	PPR, SDTR	Fast-20
0Dh - 18h	Transfer period equals the TRANSFER PERIOD FACTOR x 4	PPR, SDTR	Fast-20
19h - 31h	Transfer period equals the TRANSFER PERIOD FACTOR x 4	PPR, SDTR	Fast-10
32h - FFh	Transfer period equals the TRANSFER PERIOD FACTOR x 4	PPR, SDTR	Fast-5

Table 14 shows which transfer period factors may be used with different types of transfer agreements, provided REQ/ACK OFFSET is greater than 00h.

Table 14: Transfer period factor relationships

Value	Transfer agreement					
	Synchronous	Paced	Data group	Information unit	ST data	DT data
00h - 07h	Reserved					
08h	NS	M	NS	M	NS	M
09h	M	NS	O	NS	NS	M
0Ah	M	NS	O	NS	O	O
0Bh	M	NS	O	NS	O	O
0Ch	M	NS	O	NS	O	O
0Dh - 18h	M	NS	O	NS	O	O
19h - 31h	M	NS	O	NS	O	O
32h - FFh	M	NS	O	NS	O	O

Table abbreviations are defined as follows:

M = Mandatory: Support for the indicated transfer agreement shall be implemented if the indicated transfer period factor is implemented.

O= Optional: Support for the indicated transfer agreement may be implemented if the indicated transfer period factor is implemented.

NS = Not Supported: The indicated transfer agreement shall not be allowed if the indicated transfer factor is selected.

Table 19 defines valid combinations of Transfer Period Factor and other fields.

2.12.6 REQ/ACK offset

The REQ/ACK OFFSET field determines the maximum number of REQs allowed to be outstanding before a corresponding ACK is received at the SCSI target port during synchronous or paced transfers. For ST synchronous transfers the REQ/ACK offset is the number of REQ assertions that may be sent by the SCSI target port in advance of the number of ACK assertions received from the SCSI initiator port.

For DT synchronous transfers the REQ/ACK offset is the number of REQ transitions that may be sent by the SCSI target port in advance of the number of ACK transitions received from the SCSI initiator port. For paced transfers in DT DATA IN phase the REQ/ACK offset is the number of data valid state REQ assertions that may be sent by the SCSI target port in advance of ACK assertions received from the SCSI initiator port.

For paced transfers in DT DATA OUT phase the REQ/ACK offset is the number of REQ assertions that may be sent by the SCSI target port in advance of the number of data valid state ACK assertions received from the SCSI initiator port.

See Section 2.9 for an explanation of the differences between ST and DT data transfers.

The REQ/ACK OFFSET value is chosen to prevent overflow conditions in the port's receive buffer and offset counter. The REQ/ACK OFFSET values and which timing values shall be selected are defined in Table 15.

Table 15: REQ/ACK OFFSET timing values

Value	Description	Timing values
00h	Specifies asynchronous transfer agreement.	Asynchronous.
01h - FEh	Synchronous or paced transfers with specified offset.	Determined by transfer period factor (see Table 13).
FFh	Synchronous or paced transfers with unlimited offset.	Determined by transfer period factor (see Table 13).

Table 19 defines valid combinations of REQ/ACK OFFSET and other fields.

2.12.7 Transfer width exponent

The TRANSFER WIDTH EXPONENT field defines the transfer width to be used during DATA IN and DATA OUT phases. The values are defined in Table 16.

If any of the protocol options bits other than QAS_REQ are set to one, then only wide transfer agreements are valid. If all the protocol options bits other than QAS_REQ are set to zero, wide transfer agreements and narrow transfer agreements are valid.

Table 16: Transfer width exponent

Value	Description
00h	Specifies 8-bit data bus (i.e., narrow transfer agreement)
01h	Specifies 16-bit data bus (i.e., wide transfer agreement)
02h	Obsolete
03h - FFh	Reserved

Table 19 defines valid combinations of TRANSFER WIDTH EXPONENT and other fields.

2.12.8 Protocol options

The protocol options fields affect the protocol used between the ports. The SCSI target port uses the protocol options bits to indicate to the SCSI initiator port if it agrees to enable the requested protocol options. Except for the PCOMP_EN bit, the SCSI target port shall not enable any protocol options that were not enabled in the negotiation message received from the SCSI initiator port.

Table 17 lists the protocol options bits.

Table 17: Protocol options bits

Name	Description
PCOMP_EN	Precompensation enable.
RTI	Retain training information.
RD_STRM	Read streaming and read flow control enable.
WR_FLOW	Write flow control enable.
HOLD_MCS	Hold margin control settings.
QAS_REQ	QAS enable request.
DT_REQ	DT clocking enable request.
IU_REQ	Information units enable request.

2.12.8.1 IU_REQ

The SCSI initiator port shall set IU_REQ to one in the PPR OUT message to request that information unit transfers be enabled. In response, the SCSI target port shall set its IU_REQ to one if it agrees to use information unit transfers or zero if it does not.

The SCSI initiator port shall set IU_REQ to zero in the PPR OUT message to request that information unit transfers be disabled. In response, the SCSI target port shall set IU_REQ to zero in the PPR IN message.

If IU_REQ is one, an information unit transfer agreement is in effect. If IU_REQ is zero, an asynchronous, ST synchronous, or data group transfer agreement is in effect.

Table 19 defines valid combinations of IU_REQ and other fields.

Each SCSI target port shall maintain a bus free required flag. Each time a negotiation is successful that results in the IU_REQ bit being changed from the previous agreement (i.e., zero to one or one to zero) the SCSI target port shall set its bus free required flag to true. Any intermediate changes (e.g., from multiple successful PPR negotiations) shall be treated as changing IU_REQ even if the final value equals the initial value.

At the conclusion of the message phases, if the bus free required flag is set to true, the target port shall:

1) abort all tasks for the SCSI initiator port;

2) set the bus free required flag to false; and

3) go to a BUS FREE phase.

At the conclusion of the message phases, if the bus free required flag is set to true, the SCSI initiator device shall abort all tasks for the logical unit.

Table 18 describes the bus phases resulting from IU_REQ changes.

Table 18: Bus phases resulting from IU_REQ changes

Initial IU_REQ	Modified IU_REQ value	Causes	BUS phase following MESSAGE phases
0	0	a. PPR negotiation keeping IU_REQ set to zero; b. WDTR negotiation; or c. SDTR negotiation	COMMAND, DATA, STATUS, or BUS FREE phase
0	1	a. PPR negotiation setting IU_REQ to one	BUS FREE phase
1	0	a. PPR negotiation setting IU_REQ to zero; b. WDTR negotiation; or c. SDTR negotiation	BUS FREE phase
1	1	a. PPR negotiation keeping IU_REQ set to one	BUS FREE phase

2.12.8.2 DT_REQ

The SCSI initiator port shall set DT_REQ to one to request that DT DATA phases be enabled. In response, the SCSI target port shall set DT_REQ to one if it agrees to use DT DATA phases or zero if it does not.

The SCSI initiator port shall set DT_REQ to zero to request that information unit transfers be disabled. In response, the SCSI target port shall set DT_REQ to zero in the PPR IN message.

If DT_REQ is one, a DT data transfer agreement is in effect. If DT_REQ is zero, an asynchronous or ST data transfer agreement is in effect.

Table 19 defines valid combinations of DT_REQ and other fields.

2.12.8.3 QAS_REQ

The SCSI initiator port shall set QAS_REQ to one to request that QAS be enabled. In response, the SCSI target port shall set QAS_REQ to one if it supports QAS or zero if it does not.

The SCSI initiator port shall set QAS_REQ to zero to request that QAS be disabled. In response, the SCSI target port shall set QAS_REQ to zero in the PPR IN message.

Table 11 defines valid combinations of QAS_REQ and other fields.

When an initiator port and a target port have negotiated with each other to enable QAS, either of the two ports may participate in QAS arbitrations when attempting to connect to the other port. When an initiator port and target port have negotiated with each other to disable QAS, neither port shall participate in QAS arbitrations when attempting to connect to the other port.

When QAS and information unit transfers are both enabled for a connected SCSI target port, that SCSI target port may issue a QAS REQUEST message to release the bus after a DT DATA phase. When QAS is enabled for and information unit transfers are disabled for a connected SCSI target port, that SCSI target port shall not issue QAS REQUEST messages.

2.12.8.4 HOLD_MCS

The SCSI initiator port shall set HOLD_MCS to one to indicate that the SCSI target port should hold any margin control settings set with the margin control subpage of the port control mode page. In response, the SCSI target port shall set HOLD_MCS to one if it is capable of retaining the settings and zero if it is not.

The SCSI initiator port shall set HOLD_MCS to zero to indicate that the SCSI target port shall reset to their default values any margin control settings set with the margin control subpage of the port control mode page. In response, the SCSI target port shall set HOLD_MCS to zero.

Table 19 defines valid combinations of HOLD_MCS and other fields.

2.12.8.5 WR_FLOW

The SCSI initiator port shall set WR_FLOW to one to indicate that the SCSI target port should enable write flow control during write streaming. In response, the SCSI target port shall set WR_FLOW to one if it is capable of write flow control and zero if it is not.

The SCSI initiator port shall set WR_FLOW to zero to indicate that the SCSI target port shall disable write flow control during write streaming. In response, the SCSI target port shall set WR_FLOW to zero. Write streaming and write flow control only occurs during information unit transfers.

Table 19 defines valid combinations of WR_FLOW and other fields.

2.12.8.6 RD_STRM

The SCSI initiator port shall set RD_STRM to one to indicate that the SCSI target port should enable read streaming and read flow control. In response, the SCSI target port shall set RD_STRM to one if it is capable of read streaming and read flow control and zero if it is not.

The SCSI initiator port shall set RD_STRM to zero to indicate that the SCSI target port shall disable read streaming and read flow control. In response, the SCSI target port shall set RD_STRM to zero. Read streaming and read flow control only occur during information unit transfers.

Table 19 defines valid combinations of RD_STRM and other fields.

2.12.8.7 RTI (Retain Training Information)

The SCSI initiator port shall set RTI to one to indicate it is capable of saving paced data transfer training information and to indicate that the SCSI target port does not need to retrain on each connection. In response, the SCSI target port shall set RTI to one if it is capable of saving paced data transfer training information and zero if it is not.

The SCSI initiator port shall set RTI to zero to indicate it is not capable of saving paced data transfer training information and to indicate the SCSI target port shall retrain on each connection. In response, the SCSI target port shall set RTI to zero.

Table 19 defines valid combinations of RTI and other fields. For negotiated transfer periods slower than Fast-160 the RTI bit shall be set to zero.

2.12.8.8 PCOMP_EN

The SCSI initiator port shall set PCOMP_EN to one to indicate that the SCSI target port shall enable precompensation on all signals transmitted during DT DATA phases. The SCSI initiator port shall set PCOMP_EN to zero to indicate that the SCSI target port shall disable precompensation.

The SCSI target port shall set PCOMP_EN to one to indicate that the SCSI initiator port shall enable precompensation on all signals transmitted during DT DATA phases. The SCSI target port shall set PCOMP_EN to zero to indicate that the SCSI initiator port shall disable precompensation.

Table 19 defines valid combinations of PCOMP_EN and other fields. Ports that support Fast-160 shall support enabling and disabling precompensation of their drivers. For negotiated transfer periods slower than Fast-160 the PCOMP_EN bit shall be set to zero.

Note. Unlike other fields and bits in the PPR message the PCOMP_EN bit is not a negotiated value; instead, 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 of a successful PPR negotiation.

2.12.9 Negotiable field combinations

Not all combinations of the negotiable fields are valid. Only the combinations defined in Table 19 shall be allowed. All other combinations of the listed fields are reserved.

Table 19: Valid negotiable field combinations

Transfer period factor	REQ/ACK offset	Transfer width exponent	Protocol options								Description
			PCOMP_EN	RTI	RD_STRM	WR_FLOW	HOLD_MCS	QAS_REQ	DT_REQ	IU_REQ	
0Ah - FFh	00h	00h or 01h	0	0	0	0	0	0	0	0	Use ST DATA IN and ST DATA OUT phases to transfer data with asynchronous transfers.
0Ah - FFh	00h	00h or 01h	0	0	0	0	0	1	0	0	Use ST DATA IN and ST DATA OUT phases to transfer data with asynchronous transfers, and participate in QAS arbitrations.
0Ah - FFh	01h - FFh	00h or 01h	0	0	0	0	0	0	0	0	Use ST DATA IN and ST DATA OUT phases to transfer data with synchronous transfers.
09h - FFh	01h - FFh	01h	0	0	0	0	0	0	1	0	Use DT DATA IN and DT DATA OUT phases with data group transfers.
09h - FFh	01h - FFh	01h	0	0	0	0	0	1	1	0	Use DT DATA IN and DT DATA OUT phases with data group transfers, and participate in QAS arbitrations.
0Ah - FFh	01h - FFh	00h or 01h	0	0	0	0	0	1	0	0	Use ST DATA IN and DT DATA OUT phases with data group transfers, and participate in QAS arbitrations.
09h - FFh	01h - FFh	01h	0	0	0 or 1	0 or 1	0	0	1	1	Use DT DATA IN and DT DATA OUT phases with paced transfers and information unit transfers.
08h	01h - FFh	01h	0 or 1	0 or 1	0 or 1	0 or 1	0 or 1	0	1	1	Use DT DATA IN and DT DATA OUT phases with synchronous transfers and information unit transfers.
09h - FFh	01h - FFh	01h	0	0	0 or 1	0 or 1	0	1	1	1	Use DT DATA IN and DT DATA OUT phases with paced transfers and information unit transfers, participate in QAS arbitrations, and issue QAS_REQUEST messages to initiate QAS arbitrations.
08h	01h - FFh	01h	0 or 1	0 or 1	0 or 1	0 or 1	0 or 1	1	1	1	Use DT DATA IN and DT DATA OUT phases with information unit transfers, participate in QAS arbitrations, and issue QAS_REQUEST messages to initiate QAS arbitrations.

2.12.10 Message restrictions

PPR may be originated by SCSI initiator ports but shall not be originated by SCSI target ports. If bus expanders are present, SCSI initiator ports should only use PPR when requesting values not attainable via WDTR and SDTR (e.g., setting any protocol option bits to one). If a SCSI target port responds to PPR only with values that are attainable via WDTR and SDTR (i.e., all protocol option bits set to zero), the SCSI initiator port should repeat negotiation with a WDTR and SDTR negotiation sequence. This ensures that bus expanders that do not support PPR are still able to handle data phases correctly.

WDTR and SDTR may be originated by either SCSI target ports or SCSI initiator ports. Since WDTR resets all the values that SDTR sets, it shall be sent first if both are needed. SCSI target ports capable of wide transfer agreements shall originate negotiation with WDTR followed with SDTR.

Note. If IU_REQ was set to one and a successful SCSI target port originated WDTR negotiation occurs, a BUS FREE phase generated because the SCSI target port detected that IU_REQ was changed is indistinguishable from a BUS FREE phase generated because the target port was detecting parity errors on the WDTR OUT. Following the WDTR negotiation with an SDTR negotiation before the BUS FREE occurs ensures that the SCSI initiator port and SCSI target port both know that IU_REQ has changed.

2.12.11 Negotiation message sequences

A SCSI initiator port originated negotiation sequence contains up to four steps:

1. SCSI initiator port’s originating message;
2. SCSI target port response;
3. SCSI initiator port response; and
4. SCSI target port second response.

A SCSI target port originated negotiation sequence contains up to four steps:

1. SCSI target port’s originating message;
2. SCSI initiator port response;
3. SCSI target port response; and
4. SCSI initiator port second response.

If the negotiation fails after a vendor-specific number of retries, the SCSI port originating the negotiation sequence may discontinue communication with the other SCSI port.

For illustrations showing the various negotiation responses, refer to the ANSI SPI-4 specification.

3.0 Logical characteristics

The operations of the SCSI bus as described in this section are supported by the drive as specified in each individual drive's Product Manual, Volume 1. The drive always functions as the target unless otherwise stated.

3.1 SCSI bus phases overview

The drive responds to the following phases:

BUS FREE phase
ARBITRATION phase
SELECTION phase
RESELECTION phase

COMMAND phase
Data (IN and OUT)
STATUS (IN only)
MESSAGE (IN and OUT)

These phases are collectively termed the Information transfer phases

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

The SCSI bus can never be in more than one phase at a time. Signals that are not mentioned in a particular context shall not be asserted.

3.1.1 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 BSY and SEL are continuously negated (false) for one bus settle delay. If a SCSI device requires more than one bus settle delay to detect the BUS FREE phase, 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 a SCSI target port enters the BUS FREE phase when it releases the BSY signal.

3.1.1.1 Unexpected and expected bus free phases

In some cases a SCSI target port (connected to a SCSI initiator port) unexpectedly reverts to the BUS FREE phase to indicate an error condition that it has no other way to handle. This is called an unexpected disconnect.

SCSI target ports shall create a BUS FREE phases after any of the following:

a. after any bus reset event.

b. after a transceiver mode change reset event.

c. after an Abort Task management function is successfully received by a SCSI target port;

d. after an Abort Task Set management function is successfully received by a SCSI target port;

e. after a Clear Task Set management function is successfully received by a SCSI target port;

f. after a Logical Unit Reset management function is successfully received by a SCSI target port;

g. after a SCSI target port Reset management function is successfully received by a SCSI target port;

h. after a Clear ACA Task management function is successfully received by a SCSI target port;

i. after a DISCONNECT message is successfully transmitted from a CSI target port (see Section 4.3.2);

j. after a TASK COMPLETE message is successfully transmitted from a SCSI target port (see Section 4.3.17);

k. after a DISCONNECT message is successfully received by a SCSI target port when information unit transfers are enabled;

l. after the release of the SEL signal after a SELECTION or RESELECTION phase timeout;

m. after a PPR (Parallel Protocol Request) negotiation in response to a selection using attention condition when information unit transfers are enabled (see Section 4.3.12); or

n. after any successful negotiation that causes information unit transfers to be enabled or disabled.

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.

3.1.1.2 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 SCSI target port;

b. after a SPI data information unit is successful received by or transmitted from a SCSI target port;

c. after a SPI status information unit is successfully transmitted from a SCSI target port;

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.

3.1.2 Arbitration and QAS overview

Arbitration allows one SCSI device to gain control of the SCSI bus so that it can 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 (Quick Arbitration and Selection). 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, requires the detection of a QAS REQUEST message before starting.

SCSI devices with arbitration fairness enabled shall maintain a fairness register that records the SCSI IDs of devices that need a chance to arbitrate (see Section 3.4). Fairness in normal arbitration is enabled in targets by the Disconnect-Reconnect mode page (see Section 8.13.4). Fairness is always enabled when QAS is enabled.

3.1.2.1 Normal ARBITRATION phase

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

a. 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. 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.

b. 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.

c. Following the bus free delay in step (b), 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) 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 ANSI specification SPI-4, Annex B).

Note. There is no maximum delay before asserting the BSY signal and the SCSI ID following the bus free delay in step (b) 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.

d. After waiting at least one arbitration delay, measured from its assertion of BSY, the SCSI device shall examine the Data Bus.

   1. If no higher priority SCSI ID bit is true on the Data Bus [DB(7) is the highest], the SCSI device has won the arbitration and the SCSI device shall assert the SEL signal.
   2. If a higher priority SCSI ID bit is true on the Data Bus, the SCSI device has lost the arbitration and it shall release the BSY signal and the SCSI ID after the SEL signal becomes true (asserted), within one bus clear delay after the SEL signal becomes true. A SCSI device that loses arbitration may return to step (a). If the SCSI device implements a “fairness algorithm” for arbitration, see Section 3.4.

Note. Step (d) above requires any 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 (c). This precludes the possibility of the bus being hung.

e. After the bus free delay in step (b), 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 (b). Each SCSI device shall update its fairness register with all lower-priority device IDs that lost arbitration.

Note. For ease of implementation, this sampling may begin when BSY is true following Bus Free and end when SEL is true.

f. The SCSI device that wins arbitration shall wait at least a bus clear delay plus a bus settle delay after asserting SEL 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 of the 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.

3.1.2.2 QAS protocol

Quick Arbitration and Selection (QAS) allows a SCSI target port with an information unit transfer agreement in effect and QAS enabled (see Section 4.3.12) that is currently connected to a SCSI initiator port 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.

Before a SCSI initiator may use QAS, that initiator shall negotiate, using the PPR message, the use of the QAS phase with each SCSI target port that has indicated support of QAS. Any time a SCSI initiator port's negotiation required flag is true, that SCSI initiator port shall renegotiate to enable QAS (see Section 4.3.12).

SCSI devices that support QAS shall implement the fairness algorithm (see Annex B of SPI-4) 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 a SCSI initiator port receives a QAS REQUEST message from a SCSI target port 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.

3.1.2.3 QAS phase overview

For targets with information unit transfers and QAS enabled to indicate it wants to release the bus, the following procedure is used:

1. The target shall change to a MESSAGE IN phase, issue a single QAS Request (55h) message, and then wait for ACK to be true.

Note. The timing requirements are required to ensure that all the SCSI devices that have QAS enabled see the message bytes.

2. After detection of the ACK signal being false and if the SCSI initiator port did not create an attention condition, the SCSI target port shall release all SCSI signals except the BSY, MSG, C/D, I/O, and REQ signals. Then the SCSI target shall negate the MSG, C/D, and I/O signals within two system deskew delays. The SCSI target port shall wait two system deskew delays after negating the C/D, I/O, and MSG signals before releasing the REQ signal.
3. If the SCSI initiator port did not create an attention condition, the SCSI initiator port shall release all SCSI signals except ACK and ATN within two system deskew delays after detecting MSG, C/D, and I/O signals false. The ACK and ATN signals shall follow the timing specified in section 7 of the SPI-4 specification.

4. If the SCSI initiator creates an attention condition, the SCSI target port shall go to a MESSAGE OUT phase, receive all the message bytes, and cause an unexpected bus free by generating a BUS FREE phase (see Section 3.1.1.1 on page 53).

5. If the SCSI target port detects the SEL signal being true, the SCSI target port shall release the BSY, MSG, C/D, and I/O signals within one QAS release delay.

6. After waiting at least one QAS arbitration delay from negating the SCSI MSG, C/D, and I/O signals in step 2, if there are no SCSI ID bits true, the SCSI target port shall transition to the BUS FREE phase.

7. After waiting at least one QAS arbitration delay from negating the MSG, C/D, and I/O signals in step 2, if there are any SCSI ID bits true, the SCSI target port 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 SCSI target port shall transition to the BUS FREE phase.

Note. The release of MSG, C/D, and I/O may cause release glitches. Step 5 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 with QAS enabled to obtain control of the SCSI bus via QAS is as follows:

1. The SCSI device shall first wait for MESSAGE IN phase to occur with a single QAS REQUEST 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 a two system 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.

a. 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.

b. If a higher priority SCSI ID bit is true on the Data Bus (see Table 1 for the SCSI ID arbitration priorities) or the fairness algorithm (see Section 3.4) prevented the SCSI device from participating in QAS arbitration, then the SCSI device has lost the arbitration.

c. Any SCSI device other than the winner has lost the arbitration and shall release its SCSI ID bit after two system deskew delays and within one QAS release delay after detection of the SEL signal being asserted. A SCSI device that loses arbitration may return to step 1.

5. The SCSI device that wins arbitration shall wait at least a 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.

3.2 SELECTION phase

The SELECTION phase allows a SCSI initiator port to select a SCSI target port 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.

Refer to Section 3.4 for a description of the fairness algorithm which applies during SELECTION and RESELECTION phases.

3.2.1 Selection overview

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 a 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 a SCSI initiator port by not asserting the I/O signal.

3.2.1.1 Selection using attention condition

3.2.1.1.1 Starting the SELECTION phase when using attention condition

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, 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 that 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 timeout procedure.

The target shall not respond to a selection if bad parity is detected (see sections 3.9.2.1 and 3.9.3.1). 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.

3.2.1.1.2 Information unit transfers disabled

If information unit transfer agreement is not in effect for the connecting SCSI initiator port device, the SCSI target port shall follow the phase sequences defined in Section 3.11.

3.2.1.1.3 Information unit transfers enabled

If information unit transfers are enabled (see Section 4.3.12) for the connecting initiator, the target shall follow the phase sequences defined in Section 3.5. On detecting the MESSAGE OUT phase, the initiator shall begin a PPR (Parallel Protocol Request) negotiation (see Section 4.3.12 in this manual). 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 an SDTR message with the REQ/ACK Offset field set to 00h.

3.2.1.1.4 Selection using attention condition timeout procedure

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

- Optionally, the initiator shall assert the RST signal.
- Optionally, the initiator shall continue asserting the SEL signal and shall release the 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 signal allowing the SCSI bus to go to the BUS FREE phase. SCSI devices shall ensure that, when responding to selection, the selecting was still valid within one selection abort time of their assertion of the BSY signal. Failure to comply with this requirement may result in an improper selection (for example, two targets connected to the same initiator, wrong target connected to a SCSI initiator port or a SCSI target port connected to no initiator).

3.2.1.2 Selection without using attention condition

3.2.1.2.1 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 an 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 sections 3.9.2.1 and 3.9.3.1). Also, if more or less than two SCSI ID bits are on the Data Bus, the target shall not respond to selection.

The SCSI initiator port shall wait at least two system deskew delays after detecting that the BSY signal is true. The SCSI initiator port shall then release the SEL signal and may change the Data Bus signals. 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 Section 4.3.12) for the connecting initiator, the target shall follow the phase sequences defined in Section 3.11.

If information unit transfers are disabled (see Section 4.3.12) for the connecting initiator, the target shall follow the phase sequences defined in Section 3.12.

If a SCSI initiator port, when selecting without using an attention condition, detects an unexpected COMMAND phase, it shall set its transfer agreement to the default transfer agreement and set its negotiation required flag to true, create an attention condition, and on the corresponding MESSAGE OUT phase shall issue an ABORT TASK message. On the next selection of the SCSI target port that received the ABORT TASK message the SCSI initiator port shall do a selection using the attention condition and should negotiate to enable information unit transfers.

3.2.1.2.2 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 Section 5.3);

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, the selection was still valid within one selection abort time of their assertion of the BSY signal. Failure to comply with this requirement may result in an improper selection (for example, two targets connected to the same initiator, wrong target connected to a SCSI initiator port, or a SCSI target port connected to no initiator).

3.3 RESELECTION phase

3.3.1 RESELECTION phase overview

The RESELECTION phase allows a SCSI target port to physically reconnect to a SCSI initiator port 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.

Refer to Section 3.4 for a description of the fairness algorithm which applies during SELECTION and RESELECTION phases.

3.3.2 Physical reconnection

The SCSI device that won a normal arbitration has both the BSY and SEL signals asserted and has delayed at least a 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 SCSI target port 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 the 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 a 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 physical 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 timeout procedure.

The initiator shall not respond to a physical reconnection if bad parity is detected (see sections 3.9.2.1 and 3.9.3.1). 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. 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 ANSI SPI-4, T10/1365D.

3.3.3 Physical reconnection timeout procedure

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

- Optionally, the target shall assert the RST signal (see sections 5.3 and 5.4).
- 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 a 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 may result in an improper physical reconnection (two initiators connected to the same target or the wrong initiator connected to a SCSI target port).

3.4 SCSI bus fairness

Implementation of the SCSI bus fairness is optional, however, if implemented, the SCSI bus fairness protocol shall conform to ANSI specification SPI-4, Annex B.

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 (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 update a fairness register with the SCSI IDs of any lower priority SCSI devices that lost arbitration.

When a requirement for arbitration arises, the SCSI device shall first check to see if its fairness register is clear (see Section 3.1.2.3). If it is clear, then no lower priority SCSI devices had 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 lower level 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, or logical unit 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 the SCSI device is not arbitrating for itself, the fairness register is effectively frozen by the 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.

See ANSI specification SPI-4, Annex B, for details and timing for the SCSI bus fairness algorithm.

3.5 Information transfer phases

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

The C/D, I/O, and MSG signals are used to distinguish between the different information transfer phases (see Table 20). The target drives these three signals and therefore controls all changes from one phase to another. The initiator requests a MESSAGE OUT phase by creating an attention condition, The target causes the BUS FREE phase by releasing MSG, C/D, I/O, and BSY signals.

Table 20: 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

The information transfer phases use one or more REQ/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 transfer width (see Section 4.3.18). 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/ACK handshakes 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 SCSI target port shall not transition into an information transfer phase unless the REQ/ACK signals are negated. The target shall not transition from an information transfer phase into another information transfer phase unless the REQ and ACK signals are negated.

Note. 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. 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:

- Asynchronous transfers
- Synchronous transfers
- Paced transfers

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.

3.5.1 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 (i.e., transfer to the SCSI initiator port), the target shall first drive the DB(7-0,P_CRCA) or DB(15-0,P_CRCA,P1) signals to their desired 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 (i.e., transfer to the SCSI target port), the target shall request information by asserting the REQ signal. The SCSI initiator port 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.

3.5.2 Synchronous transfer

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 section 4.3.16 or 4.3.12). The transfer agreement specifies the REQ/ACK offset and the transfer period.

When synchronous data 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 this section for timing requirements. For timing requirements, see Section 2.5.

3.5.2.1 ST synchronous data transfer

When a ST data transfer agreement has been established, the SCSI target port 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, establishing a pacing mechanism. 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 Section 3.5.2.

If the I/O signal is true (i.e., transfer to the SCSI initiator port), 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 a 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 SCSI initiator port shall then respond with an ACK assertion.

If the I/O signal is false (i.e., transfer to the SCSI target port), the SCSI initiator port, 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 SCSI initiator port 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.

3.5.2.2 DT synchronous transfer

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/ACK transitions in this section 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, establishing a pacing mechanism. 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 Section 3.5.2.

3.5.2.2.1 Information unit transfer

When information unit transfer agreement has been established (see Section 4.3.12):

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 (i.e., 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 (i.e., 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 (i.e., transfer to the target), to transfer SPI information units, the initiator:

1. shall wait until after detecting a REQ transition;
2. shall drive the DB(15-0) signals to their desired 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 (i.e., 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;
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.

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.

3.5.2.2.1.1 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 a SCSI initiator port Detected Error message (see Section 4.3.5) to the target. This message notifies the target that data in the information unit was invalid.

If a SCSI initiator port 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 a SCSI initiator port Detected Error message (see Section 4.3.5) or an ABORT TASK message to the target. These messages notify the target that the SPI status information unit was invalid and the message received from the initiator was a SCSI initiator port Detected Error message.

If the information unit that failed was not a SPI status information unit, 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 a SCSI initiator port 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 an attention condition on or before the iuCRC is acknowledged. When the target switches to a MESSAGE OUT phase, the initiator should send a

SCSI initiator port Detected Error message (see Section 4.3.5) 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, however, 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 49, 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 Section 4.5.2) to the target. 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 for the task associated with the received ABORT TASK message.

3.5.2.2.1.2 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, however, 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 Section 3.1.1).

If a SCSI target port receives a SPI L_Q information unit with a type code that is not defined in Table 49, 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.

3.5.2.2.2 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. 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 (i.e., 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 a 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 (i.e., 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 (i.e., transfer to the target), to transfer the data field, the initiator:

1. shall wait until after 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 (i.e., 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.

3.5.2.2.2.1 Data group pad field and pCRC field transfer to SCSI initiator port

The target detects a pad field is required if the I/O signal is true (i.e., 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 without waiting for the ACK transition corresponding to the previous REQ transition unless the negotiated offset would be exceeded;
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 without waiting for the ACK transition corresponding to the previous REQ transition unless the negotiated offset would be exceeded;
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 without waiting for the ACK transition corresponding to the previous REQ transition unless the negotiated offset would be exceeded;
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 a 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;
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; and
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 transfer.

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 on or before the last transfer of the data group. When the target switches to a MESSAGE OUT phase, the initiator should send a SCSI initiator port Detected Error message (see Section 4.3.5) 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.

3.5.2.2.2.2 Data group pad field and pCRC field transfer to SCSI target port

If the I/O signal is false (i.e., transfer to the target), the initiator determines the data field. Transfer to the target is completed 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 transfer period 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 transfer.

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.

3.5.3 Paced transfer

If a paced transfer agreement has been established, it shall be used in DT DATA phase and information unit transfers shall be used. The transfer agreement also specifies the REQ/ACK offset and the transfer period.

When paced transfers are being used data shall be transferred using DT data transfers on 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 on REQ or ACK assertions, data shall be clocked by the originating SCSI device by both the assertion and negation of the REQ 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 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.

For paced DT DATA IN phases the REQ/ACK offset specifies the maximum number of data valid state REQ assertions that shall be sent by the SCSI target port in advance of the number of ACK assertions received from the SCSI initiator port. If the number of data valid state REQ assertions exceeds the number of ACK assertions by the REQ/ACK offset, the SCSI target port 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.

For paced DT DATA OUT phases the REQ/ACK offset specifies the maximum number of REQ assertions that shall be sent by the SCSI target port in advance of the number of data valid state ACK assertions that shall be sent by the SCSI target port in advance of the number of data valid state ACK assertions received from the SCSI initiator port. If the number of REQ assertions exceeds the number of data valid state ACK assertions by the REQ/ACK offset, the SCSI target port shall not assert REQ until after the next data valid state ACK assertion is received. For successful completion of a paced DT DATA OUT phase, the number of REQ assertions and data valid state ACK assertions shall be equal. Each assertion indicates a single 32-bit data transfer.

Implementors shall not use the following subsections for timing requirements. For timing requirements, see Section 2.5.

3.5.3.1 Paced transfer training pattern

If 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 after each physical connect and physical reconnect. The training pattern shall not be transferred again until after a physical disconnection occurs.

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 SCSI target port 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. The training configuration values are vendor specific.

If the retain training information is enabled and a port changes from a SCSI initiator port to a SCSI target port that SCSI target port shall retrain if the saved training configuration values were saved while the port was a SCSI initiator port.

The training pattern for a DATA IN phase shall conform to Section 3.5.3.1.1. The training pattern for a DATA OUT phase shall conform to Section 3.5.3.1.2. The receiving SCSI device shall use some or all elements of the training pattern to achieve deskewing. 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. 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.

3.5.3.1.1 DT DATA IN phase training pattern

The SCSI target port shall indicate a training pattern is going to occur on a DT DATA IN phase by:

1. releasing SEL for a minimum of two system deskew delays;
2. asserting the SEL signal a minimum of two system deskew delays; and
3. then asserting the REQ signal.

The SCSI target port shall begin the section A of its training pattern only after all the signal restrictions between information transfers phases listed in Section 3.10 or the signal restrictions between a RESELECTION phase and a DT DATA IN phase listed in Section 3.3 are met. The SCSI target port shall transmit the following training pattern:

Start of section A

1. if precompensation is enabled, then set the drivers to the strong driver state;
2. simultaneously assert REQ, P1, P_CRCA, and DB(15-0) signals;
3. wait the equivalent of 32 transfer periods (e.g., 200 ns at Fast-160);
4. simultaneously negate REQ, P1, P_CRCA, and DB(15-0) 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, P_CRCA, and DB(15-0) signals at the negotiated transfer period 64 times, (e.g., (2 x 6.25 ns) x 64 = 800 ns at Fast-160);

Start of 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. keep the P1, P_CRCA, and DB(15-0) signals negated while continuing to assert and negate REQ at the negotiated transfer period for the equivalent of 8 transfer periods (e.g., 50 ns at Fast-160);
3. keep the P1, P_CRCA, DT(15-0), and REQ signals negated for the equivalent of 8 additional transfer periods;
4. simultaneously assert and negate P1, P_CRCA, and DB(15-0) signals at twice the negotiated transfer 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 section C

1. 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 P_CRCA and DB(15-0) signals [e.g., (2 x 6.25 ns) x 64 = 800 ns at Fast-160].

The SCSI initiator port shall begin its training pattern independent of the start of the SCSI target ports training pattern if it detects the SEL, MSG, and I/O true and C/D false on the first assertion of the REQ signal. The SCSI initiator port shall transmit the following training pattern:

1. assert ACK signal within 200 ns of the first REQ assertion;
2. if precompensation is enabled then set the drivers to the strong driver state;
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);
6. set precompensation to negotiated state; and
7. assert and negate ACK signal at the negotiated transfer period 32 time, (e.g., (2 x 6.25 ns) x 32 = 400 ns at Fast-160.

At the completion of its training pattern, the SCSI target port continues asserting and negating the REQ signal at the negotiated transfer period (e.g., 6.25 ns transfer period at Fast-160) and the P1 signal at twice the negotiated transfer period (e.g., 12.5 ns transfer period at Fast-160). When the SCSI target port is ready to transfer data it shall reverse the phase of P1 (see Section 3.5.3.2).

3.5.3.1.2 DT DATA OUT phase training pattern

The SCSI target port 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 SCSI target port shall begin its training pattern only after all the signal restrictions between a SELECTION phase and a DT DATA OUT phase listed in Section 3.2.1.2 are met. The SCSI target port shall transmit the following training pattern:

1. if precompensation is enabled, set the drivers to the strong driver state;
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 the equivalent of 32 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. negate REQ and P_CRCA for at least the equivalent of 16 transfer periods (e.g., 100 ns at Fast-160); and
10. the SCSI target port shall begin asserting and negating REQ to indicate to the SCSI initiator port valid data may be sent. The number of REQ assertions shall not exceed the negotiated REQ/ACK offset.

The SCSI initiator port shall begin the section A of its training pattern independent of the start of the SCSI target ports training pattern if it detects the SEL and MSG true, and C/D and I/O false on the first assertion of the REQ signal. The SCSI initiator port shall transmit the following pattern:

Start of section A:

1. if precompensation is enabled, set the drivers to the strong driver state;
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 transfer period 64 times (e.g., (2 x 6.25) x 64 = 800 ns at Fast-160);

Start of 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. keep the P1, and DB(15-0) signals negated while continuing to assert and negate ACK at the negotiated transfer period for the equivalent of 8 transfer periods (e.g., 50 ns at Fast-160);
3. keep the P1, DB(15-0), and ACK signals negated for the equivalent of 8 additional transfer periods;
4. simultaneously assert and negate PA and DB(15-0) signals at twice the negotiated transfer 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) x 24 = 300 ns at Fast-160);

Start of section C:

1. 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 its training pattern, the SCSI initiator port continues asserting and negating the ACK signal at the negotiated transfer period (e.g., 6.25 ns transfer period at Fast-160) and the P1 signal at twice the negotiated transfer period (e.g., 12.5 ns transfer period at Fast-160). When the SCSI initiator port is ready to transfer data and the REQ/ACK offset value is not zero it shall reverse the phase of P1 (see Section 3.5.3.2 below).

3.5.3.2 P1 data valid/invalid state transitions

The transmitting SCSI device port shall indicate the start of a data valid 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 SCSI device port 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 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 SCSI device port 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 10 for examples of how the P1 signal is used to determine when the REQ or ACK transition clocks valid data.

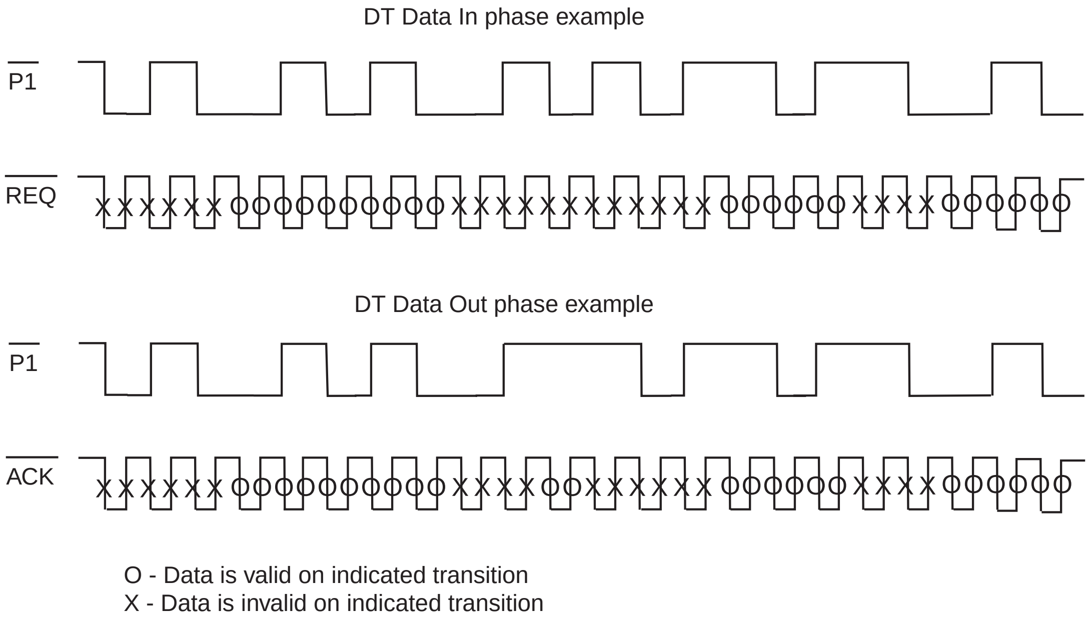

DT Data In phase example
P1
REQ
X X X X X X O O O O O O O O O O X X X X X X X X X X X X O O O O O O X X X X O O O O O O
DT Data Out phase example
P1
ACK
X X X X X X O O O O O O O O O O X X X X O O X X X X X X O O O O O O X X X X O O O O O O
O - Data is valid on indicated transition
X - Data is invalid on indicated transition

Figure 10. Use of P1 to establish data valid and data invalid states

3.5.3.2.1 Starting pacing transfers at end of training pattern

See Section 3.5.3.1 for the description of starting a data valid state after a training pattern.

3.5.3.2.2 Starting pacing transfers with no training pattern

Before starting the DT DATA IN phase, the SCSI target port shall wait at least two system 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 SCSI target port shall begin pacing transfers only after meeting all the following:

- signal restrictions between information transfer phases listed in Section 3.10;
- the signal restrictions between a RESELECTION phase and a DT DATA IN phase listed in Section 3.3.2; or
- the signal restrictions between a SELECTION phase and a DT DATA OUT phase listed in Section 3.2.1.2.

The SCSI target port shall begin pacing transfers by:

- simultaneously with the assertion of REQ, the SCSI target port shall begin asserting and negating P1 at twice the negotiated transfer period (e.g., 12.5 ns for Fast-160);

- SCSI target port shall assert and negate P1 at least 8 times [e.g., (2 x 6.25 ns) x 8 = 100 ns at Fast-160]; and
- the SCSI target port may establish a data valid state as described in Section 3.5.3.2.

The DT DATA OUT phase without training starts on the first assertion of REQ if the SEL is not asserted.

The SCSI target port shall begin pacing transfers only after meeting all the following:

- signal restrictions between information transfer phases listed in Section 3.10;
- the signal restrictions between a RESELECTION phase and a DT DATA IN phase listed in Section 3.3.2; or
- the signal restrictions between a SELECTION phase and a DT DATA OUT phase listed in Section 3.2.1.2.

The SCSI initiator port shall begin pacing transfers by:

- simultaneously with the assertion of ACK the SCSI initiator port shall begin asserting and negating P1 at twice the negotiated transfer period (e.g., 12.5 ns for Fast-160);
- SCSI initiator port shall assert and negate P1 at least 8 times (e.g., (2 x 6.25 ns) x 8 = =100 ns at Fast-160); and
- the SCSI initiator port may establish a data valid state as described in Section 3.5.3.2.

3.5.3.2.3 Ending pacing transfers

After transmitting the last data word of a DT DATA IN phase, the SCSI target port shall end pacing by waiting for all REQs to be responded to by ACKs then negate the REQ and P1 signals. After the SCSI target port stops asserting and negating REQ, it shall not assert REQ again until the requirements in Section 3.10 are met.

After transmitting the last data word of a DT DATA OUT phase, the SCSI initiator port shall:

- continue asserting and negating the ACK and P1 signals until it detects a change to the C/D, I/O, or MSG signals; and
- negate the ACK and P1 signals within 200 ns of detecting a change to the C/D, I/O, or MSG signals.

When the SCSI target port changes from a DT DATA OUT phase to any other phase it shall wait at least a bus settle delay plus a data release delay before asserting REQ and shall ignore any ACK transitions for at least a bus settle delay plus a data release delay after transitioning the C/D, I/O, or MSG signals.

3.5.3.3 Paced information unit transfer

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 (i.e., transfer to the SCSI initiator port) and the phase of the P1 signal indicates data is valid, to transfer SPI information units the SCSI target port:

- shall drive the DB(15-0) signals to their values simultaneous with the next REQ signal assertion;
- shall hold the DB(15-0) signals valid for a minimum of one transmit hold time;
- shall drive the DB(15-0) signals to their values simultaneous with the next REQ signal negation; and
- shall hold the DB(15-0) signals valid for a minimum of one transmit hold time.

If the I/O signal is true (i.e., transfer to the SCSI initiator port), to receive SPI information units the SCSI initiator port shall:

- read the value on the DB(15-0) signals within one receive hold time of the transition of the REQ signal; and
- respond with an ACK signal assertion after each REQ assertion/negation pair.

If the I/O signal is false (i.e., transfer to the SCSI target port) and the phase of the P1 signal indicates data is valid, to transfer SPI information units the SCSI initiator port:

- shall wait until detecting a REQ assertion;
- shall drive the DB(15-0) signals to their values simultaneous with the next ACk signal assertion;

- shall hold the DB(15-0) signals valid for a minimum of one transmit hold time;
- shall drive the DB(15-0) signals to their values simultaneous with the next ACK signal negation; and
- shall hold the DB(15-0) signals valid for a minimum of one transmit hold time.

If the I/O signal is false (i.e., transfer to the SCSI target port), to receive SPI information units the SCSI target port:

- shall read the value of the DB(15-0 signals within one receive hold time of the transition of the ACK signal.

If write flow control is enabled and the current SPI data stream information unit is the last SPI data stream information unit of the stream:

- the SCSI target port 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;
- the SCSI target port 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
- the SCSI target port shall negate the P_CRCA signal a minimum of a flow control transmit setup time before the start of the next information unit.

Note. The earlier in a SPI data stream information unit that the SCSI target port asserts the P_CRCA signal, the better the SCSI initiator port 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 exception handling conditions are defined in Section 3.5.2.2.1.1 and Section 3.5.2.2.1.2.

3.5.3.4 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.

3.5.4 Wide 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 wide transfer agreement is in effect (see section 4.3.18 or 4.3.12).

All SCSI devices shall support 8-bit narrow data transfers.

During narrow 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 device the DB(15-8) (if present) and DB(P1) (if present) signals are undefined.

During wide 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 21).

The IGNORE WIDE RESIDUE message may be used to indicate that the last byte of a data field or the last data byte of information unit is undefined.

Table 21: 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 device 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.

3.6 COMMAND phase

3.6.1 COMMAND phase description

The COMMAND phase allows the target to request command information from the initiator.

The SCSI target port shall assert the C/D signal and negate the I/O and MSG signals during the REQ/ACK handshakes 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.

3.6.2 COMMAND phase exception condition handling

If the target detects one or more parity errors 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.

3.7 DATA phase

3.7.1 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.

3.7.2 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 handshakes of this phase.

3.7.3 DT DATA OUT phase

The DT DATA OUT phase allows the SCSI target port to request that data be sent from the SCSI initiator device to the SCSI target device using DT data transfers. The target shall assert the MSG signal and negate the C/D and I/O signals during the REQ/ACK handshakes of this phase.

3.7.4 ST DATA IN phase

The ST DATA IN phase allows the SCSI target port 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.

3.7.5 ST DATA OUT phase

The ST DATA OUT phase allows the SCSI target device to request that data be sent from the SCSI initiator device to the target device using ST data transfers. The SCSI target port shall negate the C/D, I/O, and MSG signals during the REQ/ACK handshakes of this phase.

3.8 STATUS phase

3.8.1 STATUS phase description

The STATUS phase allows the SCSI target device to request that a status byte be sent from the target to the SCSI initiator device.

The SCSI target port 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 SCSI initiator port 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 SCSI initiator port shall assert ACK for a minimum of one QAS non-data phase REQ(ACK) period.

3.8.2 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 a SCSI initiator port Detected Error message (see Section 4.3.5) to the target. This message notifies the target that the Status byte was invalid.

3.9 MESSAGE phase

3.9.1 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.

3.9.2 MESSAGE IN phase

The MESSAGE IN phase allows the SCSI target port to request that messages be sent to the SCSI initiator port from the SCSI target port.

The SCSI target port shall assert the C/D, I/O, and MSG signals during the REQ/ACK handshakes of this phase.

A QAS-capable SCSI initiator port 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 SCSI initiator port shall assert ACK for a minimum of one QAS non-data phase REQ(ACK) period.

3.9.2.1 MESSAGE IN phase exception condition handling

If the SCSI initiator port detects a parity error on any message byte it receives, the SCSI initiator port shall create an attention condition. When the SCSI target port switches to a MESSAGE OUT phase, the SCSI initiator port shall send a MESSAGE PARITY ERROR message (see Section 4.3.7) to the SCSI target port. This message notifies the SCSI target port that the MESSAGE IN byte was invalid.

3.9.3 MESSAGE OUT phase

The MESSAGE OUT phase allows the SCSI target port 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 Section 5.1).

The target shall assert the C/D and MSG signals and negate the I/O signal during the REQ/ACK handshakes of this phase. The target shall handshake bytes 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 bytes successfully (i.e. no parity errors), it shall indicate that no retry is being attempted 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 bytes by changing to the BUS FREE phase (e.g., after receiving ABORT TASK SET or TARGET RESET messages).

3.9.3.1 MESSAGE OUT phase exception condition handling

If the target detects one or more parity errors on the message bytes received, it may request a retry of the messages 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 bytes in the same order as previously sent during this phase. When resending more than one message byte, the initiator shall reestablish the attention condition as described in Section 5.1.

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 a SCSI initiator port because of a SCSI target port detected parity error, the target shall not act on any message that it acted on the first time received.

3.10 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, SEL, and ACK signals shall not change.

b. The REQ signal shall not change until it is asserted to qualify the start of a new phase.

c. The C/D, I/O, MSG, Data Bus, and DB(P_CRCA) signals may change.

d. When switching the Data Bus or DB(P_CRCA) signal direction from out (SCSI initiator port driving) to in (SCSI target port 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.

e. The DB(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 DB(P_CRCA) signal direction from out (initiator driving) to in (target driving), the target shall delay driving the DB(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 DB(P_CRCA) signal no later than one data release delay after the transition of the C/D signal to false. When switching the DB(P_CRCA) signal direction from in (target driving) to out (initiator driving), the target shall release the DB(P_CRCA) signal no later than one system deskew delay after asserting the C/D signal. The initiator shall negate the DB(P_CRCA) signal no sooner than one system deskew delay after the detection of the assertion of the C/D signal.

f. The ATN and RST signals may change as defined under the descriptions for the attention condition (see Section 5.1) and hard reset (see Section 5.3).

3.11 SCSI bus phase sequences

3.11.1 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 a SCSI initiator port detects a phase change 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 SCSI initiator port 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 SCSI initiator ports (see Section 3.1.1).

3.11.2 Phase sequences for physical reconnection and selection using attention condition with information unit transfers disabled

The allowable sequences for either physical reconnection or selection using attention condition while an information unit transfer agreement is not in effect shall be as shown in Figure 11. The normal progression for selection using attention condition (see Section 3.2.1.1) is:

1. from the BUS FREE phase to ARBITRATION;
2. from ARBITRATION to SELECTION or RESELECTION; and
3. 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.

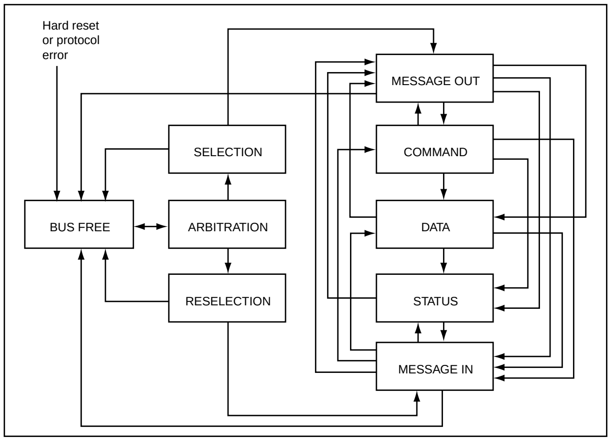


Figure 11. Phase sequences for physical reconnection and selection using attention condition with information unit transfers disabled

3.11.3 Phase sequences for selection without using attention condition with information unit transfers disabled

The allowable sequences for either physical reconnection or selection using attention condition while an information unit transfer agreement is not in effect shall be as shown in Figure 12.

The normal progression for selection without using attention condition (see Section 3.2.1.1.2) is:

1. from the BUS FREE phase to ARBITRATION;
2. from ARBITRATION to SELECTION;
3. from SELECTION to COMMAND phase;
4. from COMMAND phase to DATA phase;
5. from DATA phase to STATUS phase;
6. from STATUS phase to MESSAGE IN phase where a TASK COMPLETE message is transferred; and
7. from MESSAGE IN to BUS FREE phase.

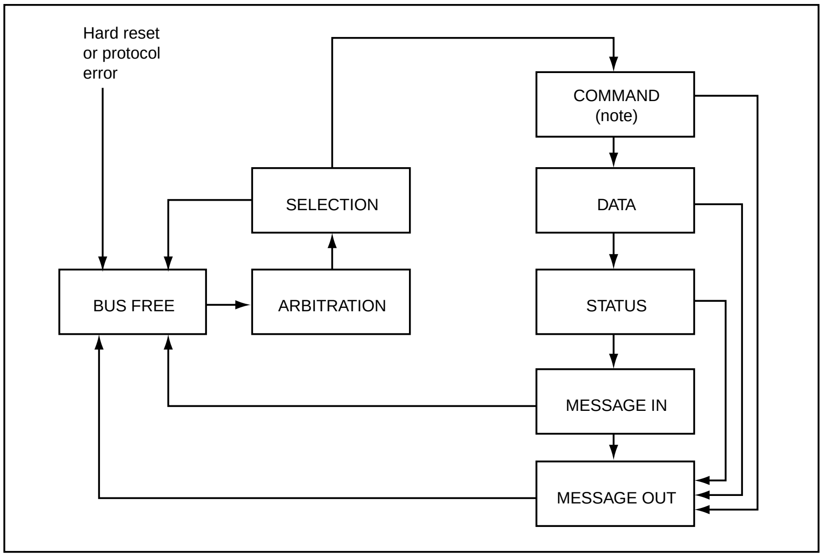


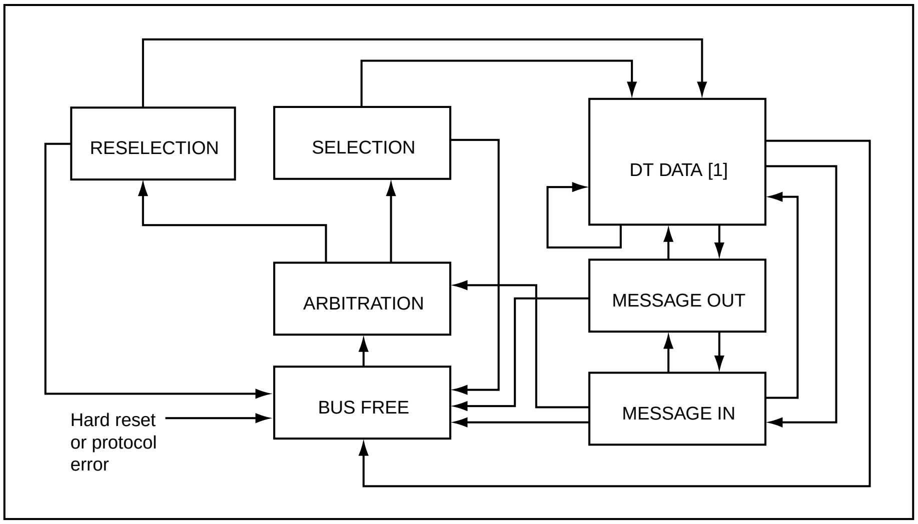
Figure 12. Phase sequences for selection without using attention condition with information unit transfers disabled

3.11.4 Phase sequences for physical reconnection or selection without using attention condition with information unit transfers enabled

The sequences for physical reconnection or selection without using attention condition while an information unit transfer agreement is in effect shall be as shown in Figure 13.

The normal progression for selection without using attention condition (see Section 3.2.1.1.3) if QAS is disabled is:

1. from the BUS FREE phase to ARBITRATION;
2. from ARBITRATION to SELECTION or RESELECTION;
3. from SELECTION or RESELECTION to one or more DT DATA phases; and
4. from the final DT DATA phase to BUS FREE.



[1] See figures 15, 16, 17, and 18 for the sequencing of SPI units within the DT DATA phases.

Figure 13. Phase sequences for physical reconnection or selection without using attention condition/ with information unit transfers enabled

3.11.5 Phase sequences for physical selection using attention condition with information unit transfers enabled

The sequences for a selection with attention condition while an information unit transfer agreement is in effect shall be as shown in Figure 14.

The normal progression for selection using attention condition (see Section 3.2.1.1.3) if QAS is disabled is:

1. from the BUS FREE phase to ARBITRATION;
2. from ARBITRATION to SELECTION;
3. from SELECTION to MESSAGE OUT;
4. from MESSAGE OUT to MESSAGE IN; and
5. from MESSAGE IN to BUS FREE.

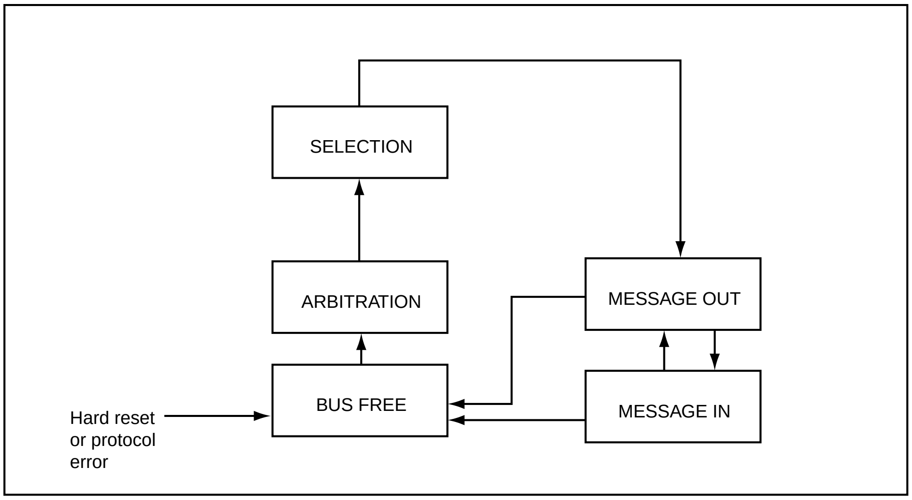


Figure 14. Phase sequences for selection with attention condition/physical reconnection and information unit transfers enabled

3.12 Data bus protection

3.12.1 Data bus protection overview

The data bus DB(P_CRCA) signal and the DB(P1) signals are used to generate parity or control the transfer of pCRC information on the Data Bus.

3.12.2 ST data bus protection using 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 the rules in Table 22.

Table 22: 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. 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.

3.12.3 DT data bus protection using CRC

3.12.3.1 DT data bus protection using 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.

3.12.3.2 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.

3.12.3.3 Order of bytes in the CRC field

Figure 81 of ANSI SCSI Parallel Interface (SPI-4), T10/1365D, Rev. 8, shows how transmitted data is used to calculate the CRC and how the CRC information is then transmitted.

4.0 Message system specification

SCSI protocol messages allow communication between a SCSI initiator port and a SCSI target port 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 initiator 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.

4.1 General message protocols and formats

One or more messages may be sent during a single MESSAGE phase, but a message shall not be split between multiple message phases.

If an information unit agreement is not in effect, the first message sent by the SCSI initiator port after a successful SELECTION phase with an attention condition shall be an IDENTIFY, ABORT TASK SET (see Section 4.5.3), or TARGET RESET message. If a SCSI target port receives any other message, it shall cause an unexpected bus free by generating a BUS FREE phase (see Section 3.1.1).

If the first message is an IDENTIFY message, 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 SCSI initiator port and the specified logical unit within the SCSI target device known as an I_T_L nexus.

If an information unit agreement is not in effect, after the RESELECTION phase the SCSI target port'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 SCSI target port 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 Section 3.1.1).

If an information unit agreement is in effect, the SCSI target port enters a DT DATA phase after the RESELECTION phase as illustrated in Figure 13.

All SCSI initiator ports shall implement the mandatory messages tabulated in the "Initiator" column of tables 26, 39, and 44. All SCSI target ports shall implement the mandatory messages tabulated in the "Target" column of tables 26, 39, and 44.

Whenever an I_T_L nexus is established by a SCSI initiator port 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.

4.2 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 23.

Table 23: Message format

Message code	Message format
00h	One-byte message (TASK COMPLETE)
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
20h–24h	Two-byte messages
25h–2Fh	Reserved two-byte messages
30h–54h	Reserved
55h	One-byte message
56–7Fh	Reserved
80h–FFh	One-byte message (IDENTIFY)

4.2.1 One-byte messages

One-byte messages consist of a single byte transferred during a MESSAGE phase. The byte's message code determines the message to be performed as defined in Tables 26, 39, and 44. The IDENTIFY message is a one-byte code, but its format is different.

4.2.2 Two-byte messages

Two-byte messages consist of two consecutive bytes transferred during a MESSAGE IN phase or a MESSAGE OUT phase. The value of the first byte determines the message to be performed as defined in Tables 26, 39, and 44. The second byte is a parameter byte used as defined in the message descriptions.

4.2.3 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. See sections referenced in Table 25 for details of the extended messages.

Table 24: Extended message format

Bit Byte	7	6	5	4	3	2	1	0
0	EXTENDED MESSAGE (01h)							
1	EXTENDED MESSAGE LENGTH (n–1) [1]							
2	EXTENDED MESSAGE CODE [2]							
3-n	EXTENDED MESSAGE ARGUMENTS [3]							

[1] The EXTENDED MESSAGE LENGTH field specifies the length in bytes of the Extended Message Code plus the extended message arguments to follow. The total length of the message is equal to the EXTENDED MESSAGE LENGTH plus two. A value of zero in the EXTENDED MESSAGE LENGTH field indicates 256 bytes follow.

[2] The extended message codes are listed in Table 25.

Table 25: Extended Message Codes

Code	Extended message
00h	MODIFY DATA POINTERS
01h	SYNCHRONOUS DATA TRANSFER REQUEST
02h	Reserved
03h	WIDE DATA TRANSFER REQUEST
04h	PARALLEL PROTOCOL REQUEST
05h	MODIFY BIDIRECTIONAL DATA POINTER
06h - FFh	Reserved

[3] The EXTENDED MESSAGE ARGUMENTS are specified within the Extended Message Descriptions in Paragraphs referenced in Table 26.

4.3 Message categories

The messages supported by the various drives are divided up into categories and listed in tables in the following subsections. The individual drive’s Product Manual, Volume 1. Table 26 lists messages that are supported by the particular drive. In the tables, the message code values are given a direction specification (In-Out). Detailed descriptions follow the tables. Messages other than those supported by a drive are answered by the drive with a MESSAGE REJECT message.

4.3.1 LINK CONTROL MESSAGES

Table 26: Link Control message codes

Code	Support				Message name	Reference section	Direction		Clear attention condition
	IU transfers disabled		IU transfers enabled						
	Init	Targ	Init	Targ			In	Out	
04h	O	O	O	O	DISCONNECT	4.3.2	In	Out	Yes
80h+	M	O	NS	NS	IDENTIFY	4.3.3	In		N/A
80h+	M	M	NS	NS	IDENTIFY	4.3.3		Out	Not required
23h	O	O	NS	NS	IGNORE WIDE RESIDUE	4.3.4	In		N/A
05h	M	M	M	M	INITIATOR DETECTED ERROR	4.3.5		Out	Yes
0Ah	O	O	NS	NS	LINKED COMMAND COMPLETE		In		N/A
09h	M	M	M	M	MESSAGE PARITY ERROR	4.3.7		Out	Yes
07h	M	M	M	M	MESSAGE REJECT	4.3.8	In	Out	Yes
01h,05h,00h	O	O	O	O	MODIFY DATA POINTER	4.3.9	In		N/A
01h,09h,05h	O	O	O	O	MODIFY BIDIREC-TIONAL DATA POINTER	4.3.9	In		N/A
08h	M	M	M	M	NO OPERATION	4.3.11		Out	Yes
01h,06h,04h	M	M	M	M	PARALLEL PROTOCOL REQUEST	4.3.12	In	Out	Yes
55h	NS	O	O	O	QAS REQUEST	4.3.13	In		N/A
03h	O	O	O	O	RESTORE POINTERS	4.3.14	In		N/A
02h	O	O	NS	NS	SAVE DATA POINTER	4.3.15	In		N/A
01h,03h,01h	O	O	M	M	SYNCHRONOUS DATA TRANSFER REQUEST	4.3.16	In	Out	Yes
00h	M	M	NS	NS	TASK COMPLETE	4.3.17	In		N/A
01h,02h,03h	O	O	M	M	WIDE DATA TRANSFER REQUEST	4.3.18	In	Out	Yes

Key:

M = Mandatory support O = Optional support NS = Not supported

IN = SCSI target port to SCSI initiator port OUT = SCSI Initiator port to SCSI target port

Yes = SCSI Initiator port shall clear the attention condition before last ACK of the Message Out phase.

Not required = SCSI Initiator port may or may not clear the attention condition before last ACK of the MESSAGE OUT phase (see Section 5.1)

N/A = Not applicable

Init = SCSI initiator port   Targ = SCSI target port

80h+ = Codes 80h through FFh are used for IDENTIFY messages

Note. The clear attention condition only applies during MESSAGE OUT phase.

4.3.2 DISCONNECT

The DISCONNECT message is sent from a SCSI target port 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 an information unit transfer agreement is not in effect, any SCSI target port 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—DISSCONNECT message sequence.

If an information unit transfer agreement is in effect, SCSI target ports 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 an information unit transfer agreement is not in effect—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 an information unit transfer agreement is in effect, 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 Section 8.13.4) 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.

4.3.3 IDENTIFY

The IDENTIFY message (see Table 27) is sent by either the initiator or the target to establish an I_T_L nexus when information unit transfers are disabled.

Table 27: 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 Section 3.8) 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 Section 3.1.1.1) 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 Section 8.13.4 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 Section 4.4 for the ordering of the IDENTIFY and TASK ATTRIBUTE messages. See Section 3.9.3.1 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 Section 4.5 for the ordering of the IDENTIFY and TASK ATTRIBUTE messages. See Section 3.9.2.1, for handling target detected errors during the MESSAGE IN phase.

4.3.4 IGNORE WIDE RESIDUE

The IGNORE WIDE RESIDUE message (see Table 28) 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 28: 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 29.

Note. More than one IGNORE WIDE RESIDUE message may occur during a task.

Table 29: Ignore field definition

	Invalid data bits
Codes	Wide transfers
00h	Reserved
01h	DB(15-8)
02h	Obsolete
03h	Obsolete
04h-FFh	Reserved

4.3.5 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 is 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.

4.3.6 LINKED COMMAND COMPLETE

The LINKED COMMAND COMPLETE message is sent from a target to an initiator to indicate that a linkied 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.

4.3.7 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 Section 3.9.3.1).

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 Section 3.1.1).

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.

4.3.8 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.

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.

4.3.9 MODIFY DATA POINTER

The MODIFY DATA POINTER message (see Table 30) is sent from the target to the initiator and requests that the signed argument be added to the value of the current data pointer using two's complement arithmetic. The data pointer is the data-out or data-in pointer being used by the command. The ENABLE MODIFY DATA POINTER (EMDP) bit in the Disconnect-reconnect mode page (see Section 8.13.4) indicates whether or not the target is permitted to issue the MODIFY DATA POINTER message. The target shall only issue the MODIFY DATA POINTER message during a unidirectional command.

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 30: 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)							
:				ARGUMENT				
6								(LSB)

4.3.10 MODIFY BIDIRECTIONAL DATA POINTER

The MODIFY BIDIRECTIONAL DATA POINTER message (see Table 31) is sent from the target to the initiator and requests that the signed data-out argument be added using two's complement arithmetic to the value of the current data-out pointer and signed data-in argument be added using two's complement arithmetic to the value of the current data-in pointer. The ENABLE MODIFY DATA POINTER (EMDP) bit in the Disconnect-reconnect mode page indicates whether the target is permitted to issue the MODIFY BIDIRECTIONAL DATA POINTER message. The target shall only issue the MODIFY BIDIRECTIONAL DATA POINTER message during a bidirectional command.

It is recommended that the target not attempt to move the data-out pointer or the data-in pointer outside the range addressed by the command. Initiators may or may not place further restrictions on the acceptable values. Should the target send a data-out argument or a data-in 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, both the data-out pointer and the data-in pointer are not changed from their values prior to the rejected MODIFY BIDIRECTIONAL DATA POINTER message.

Table 31: MODIFY BIDIRECTIONAL DATA POINTER message format

Bit Byte	7	6	5	4	3	2	1	0
0	EXTENDED MESSAGE (01H)							
1	EXTENDED MESSAGE LENGTH (09H)							
2	MODIFY BIDIRECTIONAL DATA POINTER (05H)							
3	(MSB)							
:	DATA-OUT ARGUMENT							
6								(LSB)
7	(MSB)							
:	DATA-IN ARGUMENT							
10								(LSB)

4.3.11 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.

4.3.12 PARALLEL PROTOCOL REQUEST

PARALLEL PROTOCOL REQUEST messages (see Table 32) are used to negotiate a synchronous data transfer agreement, a wide data transfer agreement, and set the protocol options between two SCSI devices.

Table 32: PARALLEL PROTOCOL REQUEST 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 OFFEST							
6	TRANSFER PERIOD EXPONENT (M)							
7[1]	PCOMP	RTI	RD_STRM	WR_FLOW	HOLD_ MCS	QAS_REQ	DT_REQ	IU_REQ

[1] For details on protocol option bits See Table 17, Section 2.12.8.

TRANSFER PERIOD FACTOR

00h-007h	Reserved. Faster timings may be allowed by future SCSI parallel interface standards.
08h	Transfer period equals 6.25 ns. Fast-160 data is latched every 6.25 ns.
09h	Transfer period equals 12.5 ns. Fast-80 data is latched every 12.5 ns.
0Ah	Transfer period equals 25 ns. Fast-40 data is latched every 25 ns or 30.3 ns.
0Bh	Transfer period equals 30.3 ns. Fast-40 data is latched every 25 ns or 30.3 ns.
0Ch	Transfer period equals 50 ns. Fast-20 data is latched using a transfer period of less than or equal to 96 ns and greater than or equal to 50 ns.
ODh - 18h	Transfer period equals the period factor x 4. Fast-20 data is latched using a transfer period of less than or equal to 96 ns and greater than orequal to 50 ns.
19h - 31h	Transfer period equals the period factor x 4. Fast-10 data is latched using a transfer period of less than or equal to 196 ns and greater than or equal to 100 ns.
32h - FFh	Transfer period equals the period factor x 4. 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.

REQ/ACK OFFSET

This field determines the maximum number of REQs allowed to be outstanding before a correesponding ACK is received at the target during synchronous or paced transfers.

For ST synchronous 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 transfers in DT DATA IN phase, the REQ/ACK OFFSET is the number of data valid state REQ assertions that may be sent by the target in advance of ACK assertions received from the initiator.

For paced transfers in DT DATA OUT phase, 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 received from the initiator.

See Section 3.5 for an explanation of the differences between ST and DT data transfers.

The REQ/ACK OFFSET value is chosen to prevent overflow conditions in the port's receive buffer and offset counter. The REQ/ACK OFFSET values and which timing values shall be selected are defined below.

Value	Description	Timing values
00h	Specifies asynchronouse transfer agreement. The Transfer period factor and protocol options other than QAS_REQ shall be ignored.	Asynch
01h-FEh	Synchronous or paced transfers with specified offset.	Determined by transfer period factor.
FFh	Synchronous or paced transfers with unlimited offset.	Determined by transfer period factor.

TRANSFER PERIOD EXPONENT

This 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 PERIOD EXPONENT field values greater than 01h are reserved.

If any of the protocol options bits (other than QAS_REQ) are set to one, then only wide transfer agreements are valid. If all the protocol options bits other than QAS_REQ are set to zero, wide transfer agreements and narrow transfer agreements are valid.

The protocol options bits (IU_REQ, DT_REQ, and QAS_REQ) 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 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. A 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.

Not all combinations of the protocol options bits are valid. Only the bit combinations defined in Table 33 shall be allowed. All other combinations are reserved.

Table 33: Valid protocol options bit combinations

QAS_REQ	DT_REQ	IU_REQ	Description
0	0	0	Use ST DATA IN and ST DATA OUT phases to transfer data
0	1	0	Use DT DATA IN and DT DATA OUT phases with data group transfers
0	1	1	Use DT DATA IN and DT DATA OUT phases with information unit transfers
1	1	1	Use DT DATA IN and DT DATA OUT phases with information unit transfers and use QAS for arbitration

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; or
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 zero, using a WDTR message with the transfer bit 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. 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. 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. The completion of an exchange of PARALLEL PROTOCOL REQUEST messages implies an agreement as shown in Table 34.

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 34: PARALLEL PROTOCOL REQUEST messages implied agreement

Target's Parallel Protocol Request response	Implied agreement
Non-zero REQ/ACK OFFSET	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).
REQ/ACK OFFSET equal to zero	Asynchronous transfer
Transfer width exponent equal to one	16-bit data
Transfer width equal to zero	Eight-bit data
Protocol options equal to 0h and transfer period factor equal to 9h	Eight-bit/asynchronous data 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
DT_REQ equal to one	DT DATA IN and DT DATA OUT phases to transfer data
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
MESSAGE REJECT message	The initiator shall set eight-bit/asynchronous data transfer with protocol options field set to 0h.
Parity error (on responding message)	Eight-bit/asynchronous data transfer with protocol options field set to 0h
Unexpected bus free (as a result of the responding message)	Eight-bit/asynchronous data transfer with protocol options field set to 0h
No response	Eight-bit/asynchronous data transfer with protocol options field set to 0h

If there is an unrecoverable parity error on the initial PARALLEL PROTOCOL REQUEST message (see Section 4.3.12), 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.

4.3.12.1 PARALLEL PROTOCOL REQUEST

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 data 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 data 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.

4.3.13 QAS REQUEST

The QAS REQUEST message is sent from a target that has both information unit transfers and QAS enabled to begin a QAS phase (see Section 3.1.2.3) after a DT DATA phase.

4.3.14 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 POINTERS message or to the value at the point at which the last SAVE DATA POINTERS message occurred for that task.

When information unit transfers are enabled, there are implied restore pointers. For more information on this see Section 6 and particularly Section 6.2.3.

4.3.15 SAVE DATA POINTERS

The SAVE DATA POINTERS 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.

4.3.16 SYNCHRONOUS DATA TRANSFER REQUEST

Synchronous Data Transfer Request (SDTR) messages (see Table 35) are used to negotiate a synchronous data transfer agreement between two SCSI devices.

Table 35: Synchronous Data Transfer Request 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 [1]							
4	REQ/ACK OFFSET							

The TRANSFER PERIOD FACTOR field values are defined below.

Code	Description
00h-07h	Reserved. Faster timings may be allowed by future SCSI parallel interface standards
08h	Transfer period equals 6.25 ns. Fast-160. DT transfers are required for this transfer period factor.
09h	Transfer period equals 12.5 ns. Fast-80. DT transfers are required for this transfer period factor.
0Ah	Transfer period equals 25 ns. Fast-40 data is latched every 25 ns or 30.3 ns.
0Bh	Transfer period equals 30.3 ns. Fast-40 data is latched every 25 ns or 30.3 ns.
0Ch	Transfer period equals 50 ns. Fast-20 data is latched using a transfer period of less than or equal to 96 ns and greater than or equal to 50 ns.
0D-18h	Transfer period equals the period factor x 4. Fast-20 data is latched using a transfer period of less than or equal to 96 ns and greater than or equal to 50 ns.
19h-31h	Transfer period equals the period factor x 4. Fast-10 data is latched using a transfer period of less than or equal to 196 ns and greater than or equal to 100 ns.
32h-FFh	Transfer period equals the period factor x 4. Fast-5 data is latched using a transfer period of less than or equal to 1,020 ns and greater than or equal to 200 ns.

The REQ/ACK OFFSET field contains 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 data 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 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.

An SDTR agreement only applies to the two SCSI devices that negotiate the agreement. Separate synchronous data transfer agreements are negotiated for each pair of SCSI devices. The synchronous data transfer agreement only applies to DATA phases.

An SDTR message exchange shall be initiated by a SCSI device whenever a previously arranged synchronous data 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 data transfer mode.

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 data 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 36.

Table 36: SDTR messages implied agreements

Responding SCSI device Parallel Protocol Request 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. This is equivalent to protocol options of 0h in the PARALLEL PROTOCOL REQUEST message.
REQ/ACK offset equal to zero	Asynchronous transfer.
MESSAGE REJECT message	The originating SCSI device shall set synchronous transfer.
Parity error (on responding message)	Asynchronous transfer
Unexpected bus free (as a result of the responding message)	Asynchronous transfer
No response	Asynchronous transfer

If there is an unrecoverable parity error on the initial SDTR message (see sections 3.12.2 and 3.12.3), 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.

4.3.16.1 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 data transfer mode 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 data transfer mode 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 data transfer mode for data transfers between the two SCSI devices.

4.3.16.2 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 data transfer mode for data transfers between the two SCSI devices.

Following a target's responding SDTR message, an implied agreement for synchronous data 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 data transfer mode for data transfers between the two SCSI devices.

4.3.17 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.

4.3.18 WIDE DATA TRANSFER REQUEST

WIDE DATA TRANSFER REQUEST (WDTR) messages (see Table 37) are used to negotiate a wide data transfer agreement between two SCSI devices.

Table 37: WIDE DATA TRANSFER REQUEST 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							

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 WDTR 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 data transfer option and uses the SDTR and WDTR messages, then it shall negotiate the wide data transfer agreement prior to negotiating the synchronous data transfer agreement. If a synchronous data transfer agreement is in effect, then:

a. If a WDTR message is rejected with a MESSAGE REJECT message, the prior synchronous data transfer agreement shall remain intact;

b. If a WDTR message fails for any other reason, the prior synchronous data transfer agreement shall remain intact; or

c. If a WDTR message is not rejected with a MESSAGE REJECT message, a WDTR message shall reset the synchronous data transfer agreement to asynchronous mode and any protocol option bits (see Section 4.3.16) shall be set to zero.

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 38.

Table 38: WDTR message implied agreement

Responding SCSI device WDTR response	Implied agreement
Transfer Width Exponent equal to one	16-bit data
Transfer Width equal to zero	Eight-bit data transfer
MESSAGE REJECT 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 sections 3.9.2 and 3.9.3) 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.

4.3.18.1 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 data 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 device-dependent 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 data transfer agreement shall remain intact.

4.3.18.2 Initiator initiated Wide Data Transfer Request (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 MESSAGE 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 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.

4.4 Task attribute messages

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 Task Attribute messages are listed in Table 39. 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 ANSI SPI-4, T10/1365D Rev. 8, Section 19.3.2).

4.4.1 Task attribute message overview and codes

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 Section 7.3). 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 may 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).

When 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 Section 7.0 of this manual and the ANSI SCSI Architecture Model-2 standard for the task set management rules.

Table 39: Task attribute message codes

Code	Support				Message name	Direction		Clear attention condition
	IU Transfers Disabled		IU Transfers Enabled					
	Iniator	**Target**	**Initiator**	**Target**				
24h	O	O	N/A	N/A	ACA		Out	Not required
21h	Q	Q	N/A	N/A	Head of Queue		Out	Not required
22h	Q	Q	N/A	N/A	Ordered		Out	Not required
20h	Q	Q	N/A	N/A	Simple	In	Out	Not required

Key:

M = Mandatory supportO = Optional supportQ = 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 Section 5.1).

N/A = Not applicable

4.4.2 ACA (AUTO CONTINGENT ALLEGIANCE)

See Table 40 for the format of the ACA message.

Table 40: 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 Section 7.0 of this manual and the SCSI Architecture Model-2 standard.

4.4.3 HEAD OF QUEUE

See Table 41 for the format of the Head of Queue message.

Table 41: 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 Section 7.0 of this manual and in the SCSI Architecture Model-2 standard.

4.4.4 ORDERED

See Table 42 for the format of the Ordered message.

Table 42: 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 Section 7.0 of this manual and in the SCSI Architecture Model-2 standard.

4.4.5 SIMPLE

See Table 43 for the format of the Simple message.

Table 43: 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 Section 7.0 of this manual and in the SCSI Architecture Model-2 standard**.**

4.5 Task management messages

Table 44 lists the codes used for messages that manage tasks. Details about these task management messages is given in subsections following the table.

4.5.1 Task management message codes

Table 44: Task management message codes

Code	Support				Message name	Direction	Clear attention condition
	IU Transfers Disabled		IU Transfers Enabled				
	Initiator	Target	Initiator	Target			
0Dh	Q	Q	M	M	ABORT TASK	Out	Yes
06h	O	M	N/A	N/A	ABORT TASK SET	Out	Yes
16h	O	O	N/A	N/A	CLEAR ACA	Out	Not required
0Eh	Q	Q	N/A	N/A	CLEAR TASK SET	Out	Yes
17h	M	O	N/A	N/A	LOGICAL UNIT RESET	Out	Yes
0Ch	O	M	O	M	TARGET RESET	Out	Yes

Key:

M = Mandatory supportO = Optional supportQ = 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 Section 5.1).

N/A = Not applicable. The receiving SCSI device shall reject this message.

4.5.2 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. 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 Section 5.1.

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. 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.

4.5.3 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.

Note. 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.

4.5.4 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.

4.5.5 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.

4.5.6 LOGICAL UNIT RESET

The LOGICAL UNIT RESET message is defined in the SCSI Architecture Model-2 standard.

Receipt of the LOGICAL UNIT RESET message after an I_T_L nexus has been astablished is a logical unit reset event as 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 LOGICAL UNIT RESET message.

4.5.7 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 shall go to the BUS FREE phase following the successful receipt of the TARGET RESET message.

5.0 Miscellaneous SCSI bus characteristics

Asynchronous conditions can occur on the SCSI bus that cause a SCSI device to perform certain actions that may alter the phase sequence of an I/O process. Furthermore, SCSI devices may not all be powered on at the same time. This publication 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. Sections 5.1, 5.3, and 5.4 describe some asynchronous conditions that could arise during operations on a SCSI bus. Section 5.5 describes SCSI pointers, a subject related to the first three. Section 4 describes the SCSI message system, which also enters into the overall SCSI operating picture.

5.1 Attention condition

The attention condition allows a SCSI initiator port to inform a SCSI target port 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 target's ID in the bus.

To create an attention condition during an information transfer phase, the initiator shall assert the ATN signal at least an attention setup time before negating the ACK signal. To reestablish 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, possibly resulting 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 26, 39, and 44. If the target detects that the initiator failed to meet this requirement, then the target shall go to BUS FREE phase (see Section 3.1).

a SCSI target port shall respond to an attention condition with MESSAGE OUT phase as follows:

a. If an attention condition is created 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 created during a DATA phase, the target shall enter MESSAGE OUT phase at the target's earliest convenience (for example, on a logical block boundary). The initiator shall continue REQ/ACK handshakes until it detects the phase change.

c. If an attention condition is created 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 created 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 created during a SELECTION phase, the target shall enter MESSAGE OUT phase after that SELECTION phase.

f. If SPI (SCSI Parallel Interface) information unit transfers are disabled and an attention condition is detected during a RESELECTION phase, the SCSI target port shall enter MESSAGE OUT phase after the SCSI target has sent its IDENTIFY message for that RESELECTION phase.

g. If an attention condition is created during a RESELECTION phase the target shall enter MESSAGE OUT phase after the target has sent the first Information Transfer phase.

h. If the attention condition is detected during the transfer of a SPI data stream information unit, the target shall terminate the current stream by entering into MESSAGE OUT phase at the end of any SPI data stream information unit in the current stream.

i. 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.

5.2 Bus reset condition

The bus reset condition is used to create a hard reset (see Section 5.3) for all SCSI devices on the bus and change the bus to a BUS FREE phase. This condition shall take precedence over all other phases and conditions. Any SCSI device may create the bus reset condition by asserting the RST signal for a minimum of a reset hold time.

Environmental conditions (for example, 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. This limits the time to confirm the RST signal to a maximum of a bus clear delay.

5.3 Hard reset

A SCSI device detecting a reset event shall release all SCSI bus signals within a bus clear delay of the transition of the RST signal to true, except the RST signal if it is asserting RST. The BUS FREE phase always follows the hard reset condition. The SCSI device shall not assert the RST signal in response to a reset event on the same bus segment.

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.

Any SCSI device that detects a hard reset shall also set its transfer agreement to the default transfer agreement.

5.4 Reset events

When a SCSI device detects a reset event it shall initiate hard reset.

5.4.1 Bus reset event

When a SCSI device detects a bus reset condition by detecting RST true for a reset delay, it shall cause a reset event. In response to a bus reset event, a SCSI target port shall create a unit attention condition for all initiators. The sense key shall be set to Unit Attention with the additional sense code set to either SCSI Bus Reset Occurred or Power On, Reset, or Bus Device Reset Occurred.

5.4.2 Power on reset event

When a SCSI device is powered on, it shall cause a reset event. In response to a power on reset event, the target shall create a unit attention condition for all initiators. The sense key shall be set to Unit Attention with the additional sense code set to either Power On Occurred or Power On, Reset, or Bus Device Reset Occurred.

5.4.3 Target reset event

When a SCSI device successfully receives a SCSI target port Reset message, it shall cause a reset event. In response to a SCSI target port reset event, the SCSI target shall create a unit attention condition for all SCSI initiator ports. The sense key shall be set to Unit Attention with the additional sense code set to either Bus Device Reset Function Occurred or Power On, Reset, or Bus Device Reset Occurred.

5.4.4 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 SCSI target port 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 Single-ended.

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 SCSI target port 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.

5.5 Asynchronous condition recovery

Part of the asynchronous condition recovery stem are sets of pointers to storage areas where information is kept that, in addition to being necessary to the start of a task, allows tasks to be interrupted, restarted or repeated. The following subsections describe these pointers.

5.5.1 SCSI pointers

The SCSI architecture provides for sets of pointers to be kept within each initiator task control area. These pointers are in sets of three pointers per set. The pointers in each set point to three storage area sections in the initiator. The three sections contain the following information:

a. A command from initiator to target.
b. Status (from target) associated with the command.
c. Data (to/from target) associated with the command.

Of these three-pointer sets there are two types:

- Active pointers (one set per initiator only).
- Saved pointers (one or more sets per initiator, up to fifteen sets total).

The use of these two types of pointers is described in the following paragraphs.

5.5.2 Active pointers

Active pointers represent the current state of the interface between the initiator and the target that the initiator is currently connected to and servicing. The pointers for the current task remain in the initiator Active Pointer registers from the time they are put there after the completion of all activities associated with the previous task, until the logic of the Initiator dictates a new task shall be executed. Normally, successful receipt by the initiator of good status associated with the current task triggers the initiator to insert a new set of the three Active pointers for the next task. The initiator does not wait for the TASK COMPLETE message before deciding whether to retry the Active task or transfer in new task pointers. If the current task was never satisfactorily completed, the initiator logic may dictate that some special response action be taken, such as restoring the values in the Active pointer registers to their beginning values so the current task can be restarted (see note following). The initiator may choose to send a command such as a REQUEST SENSE command to the target, or the initiator could ignore the unsatisfactorily completed task and start the next originally scheduled task.

Note. For example, if the drive detects a parity error in the data out from the current command, it sends the RESTORE POINTERS message to the initiator. In this case, the Restore Pointers request causes the initiator to restore as the active pointers the values existing at the beginning of the current task so the current task can be resent. The “beginning” pointer values point to the first byte of the current task (very likely a command descriptor block), the first byte of the area set aside for status to be returned and the first byte of the area set aside for data associated with the current task. (See Section 4.3.14 for a detailed description of operations resulting from the RESTORE POINTERS message).

5.5.3 Saved pointers

Saved pointers point to initiator storage locations where command, status and data information are stored for a task that was saved at some point in the past. There is one set of saved pointers for the current task for each target on the interface bus that is currently active (whether or not it is currently connected to the initiator). The saved command pointer always points to the first byte of the Command Descriptor Block (see Section 7.2) for the “current” task for each target, the saved status pointer always points to the first byte of the area used for the status associated with the command, and the saved data pointer points to some location (not necessarily the beginning) in the area used for data associated with the command. When a SCSI target port disconnects the initiator saves the current pointers. Before a SCSI target port disconnects it may send a SAVE DATA POINTERS message to the initiator, which copies the data pointer that is for the current command for that target into the location set aside for the target’s saved pointers. When a SCSI target port reconnects, the initiator performs a restore pointers operation that copies the saved pointers for the reconnected target into the initiator current pointer registers so that the current command for that target may continue its operation from where it left off before disconnecting. If the target had sent a Save Data Pointer message previously, the current data pointer points to the place in the data store area where operations left off. The data pointer otherwise points to the beginning of the data area, unless the data pointers were modified by a MODIFY DATA POINTERS message from the target prior to disconnecting. The MODIFY DATA POINTERS message adds a value to the data pointer that allows data to be taken, upon reconnection, from a location before or after the last byte transferred location.

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 for that purpose.

5.6 Command processing considerations and exception conditions

5.6.1 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.

5.6.2 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 SCSI target port 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, Section 4.3.16, 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. 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.

This feature is not likely supported at this time. Check individual drive’s Product Manual, Volume 1, in features supported table to see if it is supported by a particular drive model.

5.6.3 Incorrect initiator connection

An incorrect initiator connection occurs during an initial connection if a SCSI initiator port 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 Section 4.4).

Note. 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).

5.6.4 Unexpected RESELECTION phase

An unexpected RESELECTION phase occurs if a SCSI target port attempts to do a physical reconnect to a task for which a nexus does not exist. a SCSI initiator port should respond to an unexpected RESELECTION phase by sending an ABORT TASK message.

6.0 SPI information units

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 figures 15, 16, 17, and 18. See figures 12 and 13 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 sections 3.5.2.2.1.1 and 3.5.2.2.1.2.

The normal progression is from SPI L_Q information unit/SPI command information unit pairs, to SPI L_Q information unit/SPI data information unit pairs, to SPI L_Q information unit/SPI status information unit pairs.

Note. a SCSI initiator port may request a BUS FREE phase by creating an attention condition and sending a DISCONNECT message on the corresponding MESSAGE OUT phase. This allows a SCSI initiator port to request the target break up a long sequence of SPI L_Q information unit/SPI data information unit pairs into smaller sequences.

After message phases complete that contain any negotiation (such as, PPR, WDTR, or SDTR) that results in UI_REQ being changed, the SCSI target device shall abort all tasks, except the current task, for the initiator participating in the negotiation and the SCSI initiator shall abort all tasks, except the current task, for the target device.

When an information unit transfer agreement is in effect, 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.

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 Section 6.2.4).

6.1 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 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 SPI L_Q information unit Data Length field is zero; and

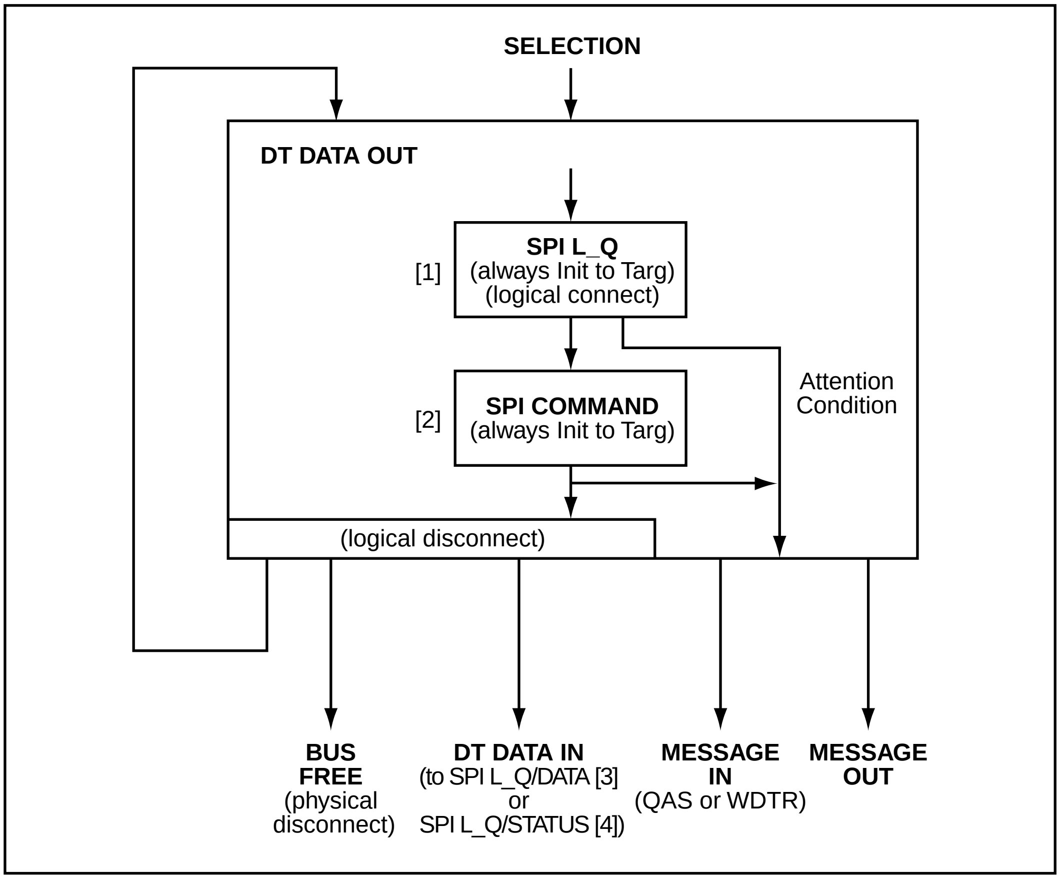
e. the last SPI data stream information unit.

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 a SCSI initiator port 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.



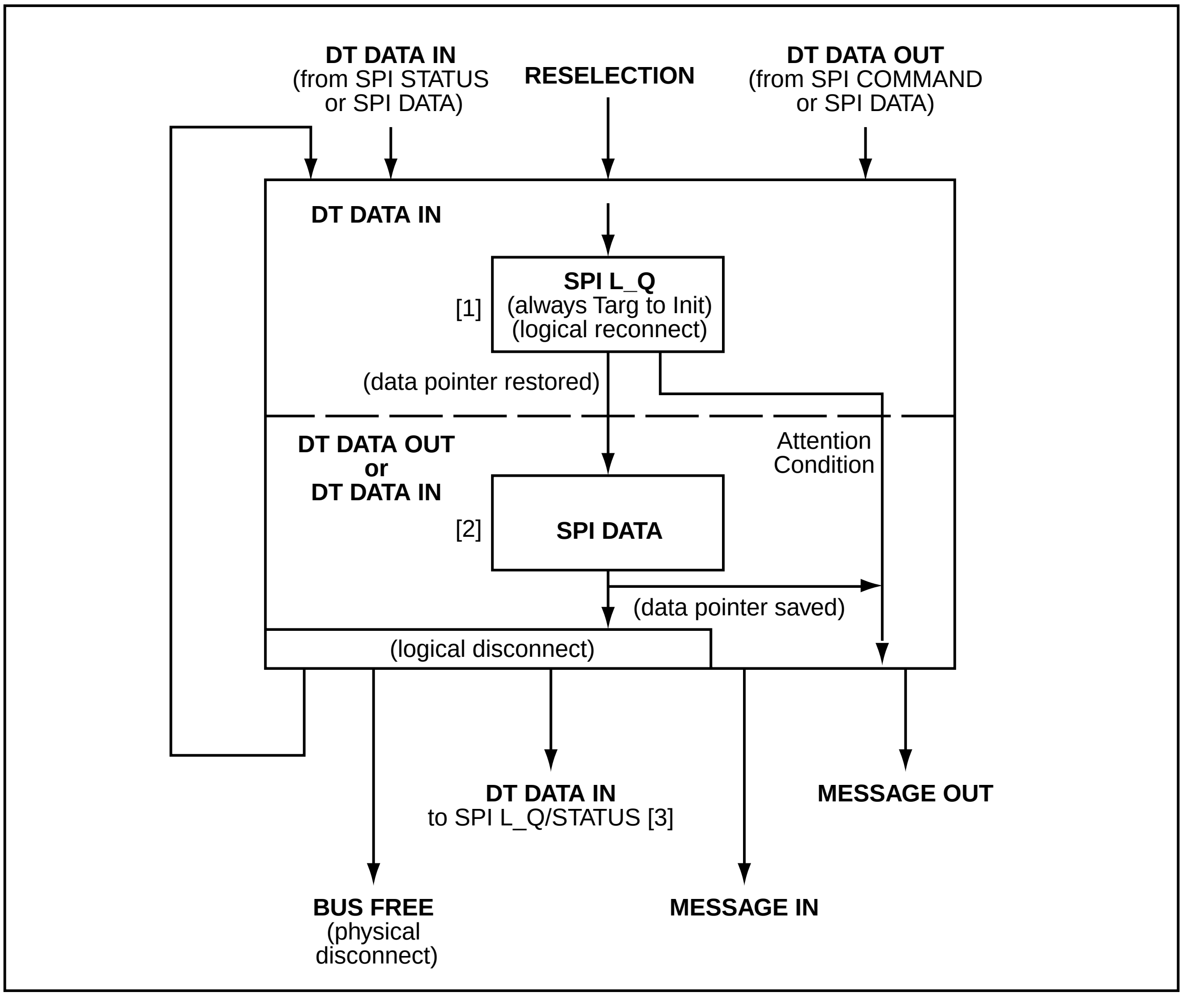
[1] See Table 48 for the format of the SPI L_Q information unit.

[2] See Table 45 for the format of the SPI Command information unit.

[3] See Figure 16.

[4] See Figure 18.

Figure 15. SPI information unit sequence during initial connection

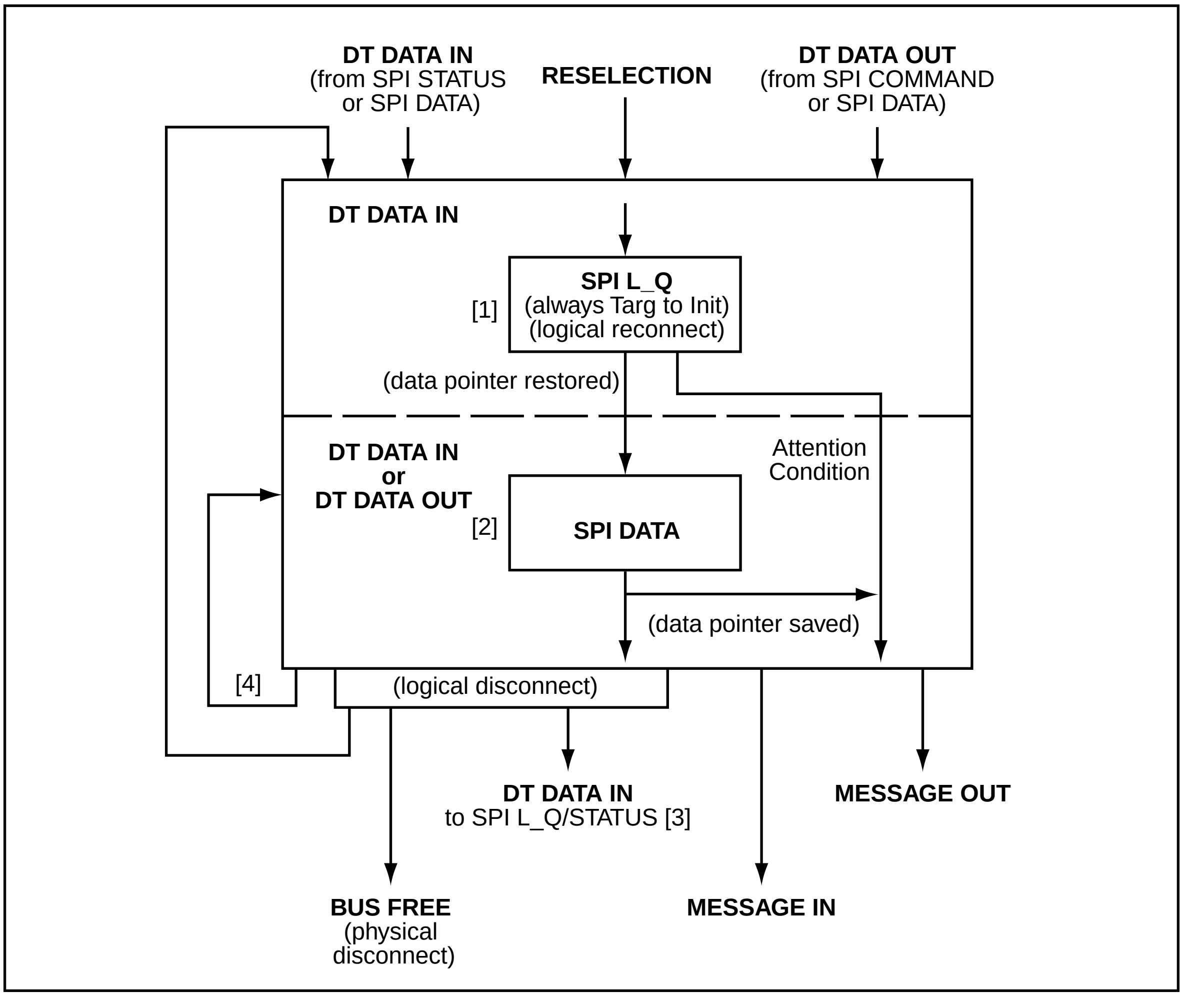


[1] See Table 48 for the format of the SPI L_Q information unit.

[2] See Table 51 for the format of the SPI Data information unit.

[3] See Figure 18.

Figure 16. SPI information unit sequence during data type transfers

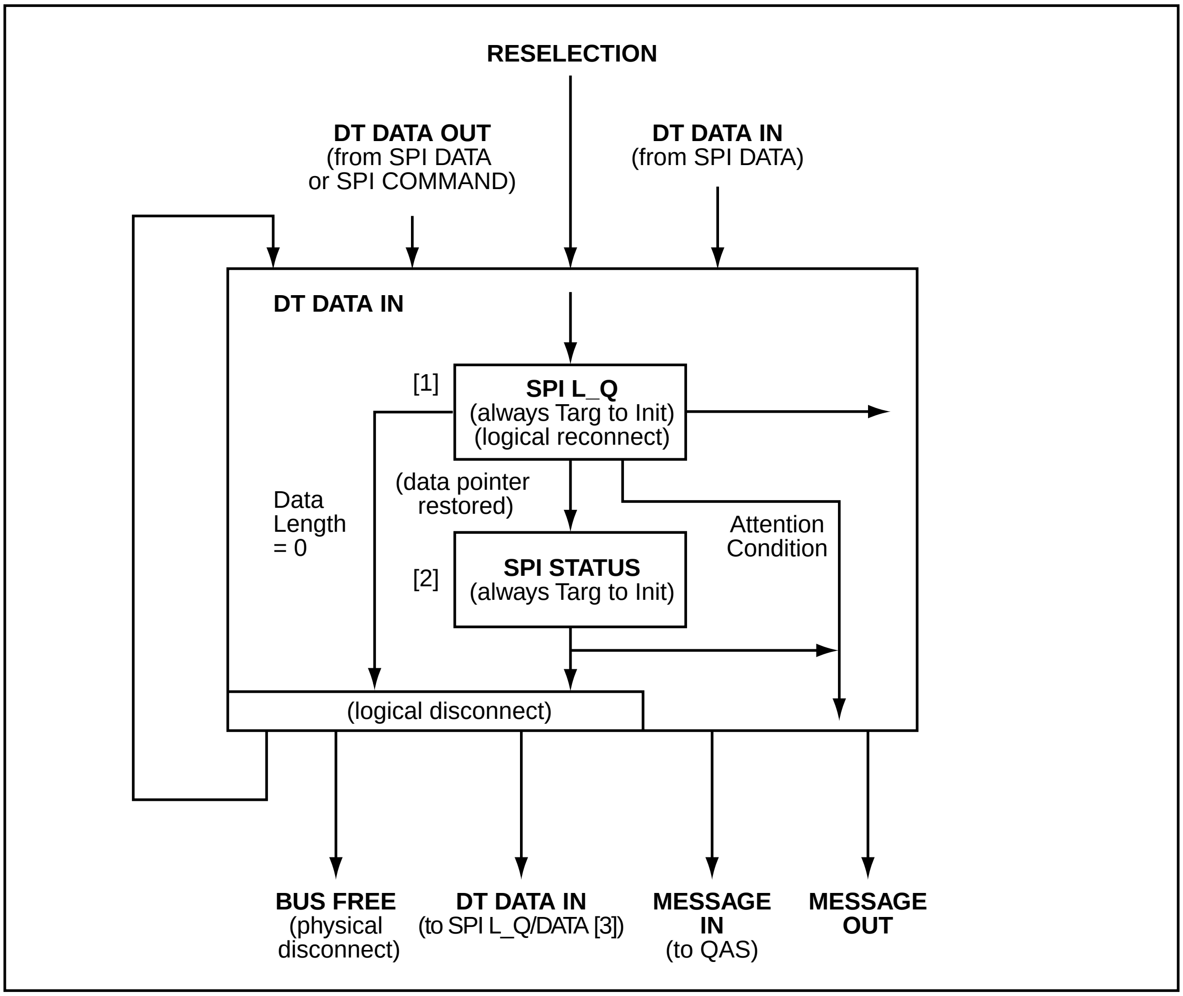


[1] See Table 48 for the format of the SPI L_Q information unit.

[2] See Table 52 for the format of the SPI Data information unit.

[3] See Figure 18.

Figure 17. SPI information unit sequence during data stream type transfers



[1] See Table 48 for the format of the SPI L_Q information unit.

[2] See Table 53 for the format of the SPI Status information unit.

[3] See Figure 16.

Figure 18. SPI information unit sequence during status transfers

6.2 SPI information units

6.2.1 SPI command information unit

The SPI command information unit (see Table 45) transfers CDBs, task attributes, and task management requests to be performed by a target.

a SCSI initiator port 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 SCSI target port 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 but need not hold the transmitted information. After transferring all the SPI command information unit bytes the target shall change to a DT DATA IN phase and transmit a SPI status information unit with the status defined in the SCSI Architecture Model-2 standard (SAM-2) for this condition. If the initiator has more commands to send 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 to 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 support 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 45: SPI command information unit

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved					Task Attribute [1] [7]		
2	Task Management Flags [2] [7]							
3	Additional CDB Length = (number of additional CDB bytes)/4 [3]						RDDATA [4]	WRDATA [4]
4	(MSB)	CDB [5] [7] [8]						
19								(LSB)
20	(MSB)	Additional CDB [6] [7] [8]						
n								(LSB)
n+1	(MSB)	iuCRC [9]						
n+2								
n+3								
n+4								(LSB)

[1] The Task Attribute field is defined in Table 46.

Table 46: Task Attribute

Codes	Description
000b	Requests that the task be managed according to the rules for a simple task attribute. See the SCSI Architecture Model-2 standard.
001b	Requests that the task be managed according to the rules for head of queue task attribute. See the SCSI Architecture Model-2 standard.
010b	Requests that the task be managed according to the rules for an ordered attribute. See the SCSI Architecture Model-2 standard.
011b	Reserved.
100b	Requests that the task be managed according to the rules for a simple task attribute. See the SCSI Architecture Model-2 standard.
101b- 111b	Reserved.

[2] The Task Management Flags field is defined in Table 47. If a Task Management function fails, the Task Manager shall terminate the task with Good status. The packetized failure code shall be set to Task Management Function Failed.

Table 47: 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 Section 4.5.2).
02h	The task manager shall abort the task set as defined in the ABORT TASK SET message (see Section 4.5.3).
04h	The task manager shall clear the task set as defined in the CLEAR TASK SET message (see Section 4.5.5).
08h	The task manager shall perform a hard reset to the selected logical unit as defined in the LOGICAL UNIT RESET message (see Section 4.5.6).
20h	The task manager shall perform a hard reset as defined in the TARGET RESET message (see Section 4.5.7).
40h	The task manager shall perform a clear ACA as defined in the CLEAR ACA message (see Section 4.5.4).
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.

[3] The Additional CDB Length field contains the length in 4-byte words of the Additional CDB field.

[4] The Write Data (WRDATA) bit and READ Data (RDDATA) bit are defined in SCSI protocol standards T10/1144D (FCP-2), sections 9.1, 9.1.16, and 9.1.17. They are also discussed in the Seagate *Fibre Channel Interface Manual*, part number 77767496. These two bits are used by drives in a Fibre Channel I/O system and shall be ignored by this section.

[5] The CDB field contains the actual CDB to be interpreted by the addressed logical unit. The maximum CDB length is 16 bytes. 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.

[6] The Additional CDB field contains any CDB bytes beyond those contained within the standard 16 byte CDB field.

[7] The CDB, Additional CDB, and Task Attribute fields are not valid and are ignored if the Task Management Flags field is not zero.

[8] The contents of the CDB and Additional CDB fields shall be as defined in the SCSI command standards.

[9] The iuCRC field shall use the algorithm defined in Subsection 11.3 of ANSI SCSI Parallel Interface (SPI-4).

6.2.2 SPI L_Q information unit

The SPI L_Q information unit (see Table 48) contains L_Q nexus (Logical unit—Q tag relationship) information for the information unit 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 a SCSI initiator port shall cause the initiator to restore the data pointer.

Table 48: SPI L_Q information unit

Bit Byte	7	6	5	4	3	2	1	0
0	Type [1]							
1	Reserved							
2	MSB	Tag [2]						
3								LSB
4	MSB	Logical Unit Number [3]						
5								
6								
7								
8								
9								
10								
11								LSB
12	Reserved							
13	MSB	Data Length [4]						
14								
15								LSB
16	BIDI Direction [5]		Reserved					
17	Reserved							
18	MSB	iuCRC Interval [6]						
19								LSB
20	MSB	iuCRC [7]						
21								
22								
23								LSB

[1] The Type field is defined in Table 49. If a SCSI initiator port receives a type code that is not defined in Table 49, that initiator shall follow the procedures defined in Section 3.5.2.2.1.1. If a SCSI target port receives a type code that is not defined in Table 49, that target shall follow the procedures defined in Section 3.5.2.2.1.2.

Table 49: SPI L_Q information unit type

Codes	Type	Description
01h	Last command	Sent by a SCSI initiator port device to indicate a SPI command information unit shall follow this SPI L_Q information unit. Indicates the initiator device 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 device. The Bidi Direction field shall be set to zero and ignored by the receiving SCSI target device.
02h	Multiple command	Sent by a SCSI initiator port device to indicate a SPI command information unit shall follow this SPI L_Q information unit. Indicates the initiator device 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 SCSI target device. The IUCRC Interval field shall be set to zero and ignored by the SCSI target device. The BIDI Direction field shall be set to zero and ignored by the receiving target device.
04h	Data	Sent by a SCSI target port device 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. For a bidirectional command, the direction of the SPI data information unit shall be indicated in the BIDI Direction field of the SPI L_Q information unit as defined in Table 50.
05h	Data stream	Sent by a SCSI target port device to indicate an unspecified number of SPI data stream information units shall follow this SPI L_Q information unit. The Data Length field shall not be set to zero. For a bidirectional command, the direction of the SPI data stream information units shall be indicated in the BIDI Direction field of the SPI L_Q information unit as defined in Table 50.
08h	Status	Sent by a SCSI target port device 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 Section 6.2.5). The iuCRC Interval field shall be set to zero and ignored by the SCSI target device. The BIDI Direction field shall be set to zero and ignored by the receiving SCSI initiator device.
F0H - FFh		vendor-specific.
All others		Reserved.

[2] Tag field is a 16-bit unsigned integer assigned by the initiator and sent to the initiator in the Send SCSI Command request. See Section 4.4.1, Task attribute message overview and codes.

[3] 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, Section 4.11. 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-3 (SPC-3) standard. The eight bytes in this field provide 4 levels of addressing. Bytes 4-5 address the first level, etc. to bytes 10-11, which address the fourth level. If LUN in SPI LQ information unit packet is X which is greater than 0x3F (>63), then the LUN in ID message sent back will be set to 0x3F (63). If LUN in SPI LQ information unit packet is X which is not greater than 0x3F (>63), then the LUN in ID message sent back will be set to X.

[4] The Data Length field contains the length in bytes of the following information units. 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 Section 8.13.4).

[5] The BIDI Direction field determines the data direction if the command is a bidirectional command and the type code is data or data stream. The code values for the BIDI Direction field are defined below.

Table 50: BIDI Direction

Codes	Description
00b	A unidirectional command or a type code other than data or data stream (see Table 49).
01b	A bidirectional command transferring data from the SCSI initiator device to the SCSI target device.
10b	A bidirectional command transferring data from the SCSI target device to the SCSI initiator device.
11b	Reserved.

[6] 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.

[7] The iuCRC field shall use the algorithm defined in Subsection 11.3 of the ANSI SCSI Parallel Interface (SPI-4).

6.2.3 SPI data information unit

The SPI data information unit (see Table 51) contains data.

The detection of a BUS FREE phase following a SPI data information unit by a SCSI initiator port shall be equivalent to the initiator receiving a DISCONNECT message.

The detection of a QAS REQUEST message following a SPI data information unit by a SCSI initiator port shall be equivalent to the initiator receiving a DISCONNECT message.

Table 51: SPI data information unit

Bit Byte	7	6	5	4	3	2	1	0
0	MSB	Data [1]						
n								LSB
n+1	MSB	iuCRC [2]						
n+2								
n+3								
n+4								LSB

[1] The Data field may contain any type of information (e.g., parameter lists, mode pages, user data).

[2] The iuCRC field shall use the algorithm defined in Subsection 11.3 of ANSI SCSI Parallel Interface (SPI-4), T10/1365D. 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 this table. These additional iuCRC fields are not shown in this table.

6.2.4 SPI data stream information unit

The SPI data stream information unit (see Table 52) contains data.

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 SCSI 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 nits a SCSI 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 SCSI target shall end a sequence of SPI data stream information units by performing one of the following:

a. should assert the P_CRCA signal before the end of the current SPI data stream information unit boundary; or

b. may change the C/D, I/O, or MSG signals on a SPI data stream information unit boundary.

If, during the last SPI data stream information unit of a read stream, the P_CRCA signal was not asserted and a SCSI 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 asserted and a SCSI initiator port detects a REQ transition after receiving the last iuCRC for a SPI data stream information unit, that SCSI initiator shall logically disconnect from the current I_T_L_Q nexus.

If, during a sequence of SPI data stream information units a SCSI initiator detects 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 a SCSI initiator shall be equivalent to the initiator port receiving a DISCONNECT message.

The detection of a QAS REQUEST message following a SPI data stream information nit by a SCSI initiator shall be equivalent to the initiator receiving a DISCONNECT message.

To end a sequence of SPI data stream information units, a SCSI initiator may request a disconnect by establishing an attention condition. The initiator shall continue to transfer or receive data, pad bytes (if any), and iuCRCs until the target changes to the MESSAGE OUT phase.

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.

Table 52: SPI status data stream information unit

Bit Byte	7	6	5	4	3	2	1	0
0	MSB			Data [1]				
n								LSB
n+1	MSB			iuCRC [2]				
n+2								
n+3								
n+4								LSB

[1] The Data field may contain any type of information (e.g., parameter lists, mode pages, user data).

[2] The iuCRC field shall use the algorithm defined in Subsection 11.3 of ANSI SCSI Parallel Interface (SPI-4). 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 this table. These additional iuCRC fields are not shown in this table.

6.2.5 SPI status information unit

The SPI status information unit (see Table 53) 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 Section 6.2.2) to zero.

Table 53: SPI status information unit

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved							
2	Reserved				Reserved for FCP [1]		SNSVALID [2]	RSPVALID [4]
3	Status [5]							
4	MSB	Sense Data List Length (n-m) [3]						
7								LSB
8	MSB	Packetized Failures List Length [4]						
11								LSB
12	MSB	Packetized Failures [4]						
m								LSB
1+m	MSB	Sense Data List [3]						
n								LSB
n+1	MSB	iuCRC [6]						
n+2								
n+3								
n+4								LSB

[1] Reserved for Fiber Channel Protocol.

[2] A Sense Data Valid (SNSVALID) bit 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.

[3] If sense data is provided, the Sense Data Valid (SNSVALID) bit 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 shall be set to zero. The initiator shall ignore the Sense Data List Length field and shall assume a length of zero.

[4] If packetized failure data is provided, the Packetized Failures Valid (RSPVALID) bit 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 shall be set to zero. The initiator shall ignore the Packetized Failures List Length field and shall assume a length of zero. The Packetized Failures field (see Table 54) 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 55).

[5] The Status field contains the status of a task that completes. See Section 7.3, Table 62, for a list of status codes.

[6] The iuCRC field shall use the algorithm defined in Subsection 11.3, ANSI SCSI Parallel Interface (SPI-4), T10/1365D.

The Packetized Failures field (see Table 54) 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 55).

Table 54: Packetized Failures field

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved							
2	Reserved							
3	Packetized Failure Code [1]							

[1] The Packetized Failures Code is defined in Table 55.

Table 55: Packetized Failures code

Codes	Description
00h	Indicates no failure or task management function complete.
01h	Reserved.
02h	SPI command information unit fields invalid.
03h	Reserved.
04h	The task management function not supported.
05h	The task management function failed.
06h-FFh	Reserved.

The Sense Data field contains the information specified by the SCSI Primary Commands-3 (SPC-4), T10/ 1416D, for presentation by the REQUEST SENSE command (see Section 8.38). The proper sense data shall be presented when a SCSI status byte of Check Condition is presented as specified by the SCSI Primary Commands-3 (SPC-3), T10/1416D and this manual.

The iuCRC field shall use the algorithm defined in Subsection 11.3, ANSI SCSI Parallel Interface (SPI-4), T10/ 1365D.

7.0 SCSI commands

This section defines the SCSI command structure and describes a typical SCSI bus procedure involving a command, status return, and message interchange.

The command structure defined herein provides for a contiguous set of logical blocks of data to be transferred across the interface. The number of logical data blocks to be transferred is defined in the command. Initiator commands to the drive are structured in accordance with the requirements imposed by the drive physical characteristics. These physical characteristics are reported to the initiator in response to an INQUIRY command.

A single command may transfer one or more logical blocks of data. The drive may disconnect, if allowed by the initiators, from the SCSI bus to allow activity by other SCSI devices while the drive performs operations within itself.

Upon command completion (which may be executed either successfully or unsuccessfully), the drive returns a status byte to the initiator. Since most error and exception conditions cannot be adequately described with a single status byte, one status code that can be sent as the status byte is called Check Condition. It indicates that additional information is available. The initiator may issue a REQUEST SENSE command to request the return of the additional information as part of the DATA IN phase of the command.

Future implementations of the SCSI protocol will contain an autosense feature (see Section 7.6.4.2) allowing the initiator to request the automatic return of sense data. Fibre Channel protocol already has this feature in its packetized command structure.

7.1 Command implementation requirements

The first byte of any SCSI command contains an operation code as defined in this document. Three bits (bits 7-5) of the second byte of each SCSI command have historically been used to specify the logical unit if it is not specified using the Identify Message (see Section 4). These three bits are now shown as “Reserved” in the new Command data block format. The last byte of most SCSI commands contains a control field. The exception to this is the new variable length CDB which has the control field in the second byte, rather than the last.

7.1.1 Reserved

Reserved bits, bytes, fields, and code values are set aside for future standardization. Their use and interpretation may be specified by future revisions to this specification. A reserved bit, field, or byte shall be set to zero, or in accordance with a future revision to this specification. A drive that receives a reserved code value shall terminate the command with a CHECK CONDITION status and the Sense Key shall be set to ILLEGAL REQUEST. It shall also be acceptable for the drive to interpret the bit, field, byte, or code value in accordance with a future revision to this specification.

7.2 Command Descriptor Block (CDB)

A request by a SCSI initiator port to a drive is performed by sending a Command Descriptor Block (CDB) to the drive. For several commands, the request is accompanied by a list of parameters sent during the DATA OUT phase. The field uses shown in tables 56, 57, 58, 59, and most of those in 60 are used consistently by most commands. See the specific commands for detailed information.

The Command Descriptor Block always has an operation code as the first byte of the command. This is followed by command parameters (if any), and a control field.

For all commands, if there is an invalid parameter in the Command Descriptor Block, the drive shall terminate the command without altering the medium.

The format description for the Command Descriptor Block as supported by Seagate drives is shown in tables 56, 57, 58, 59, and 60.

7.2.1 Fixed and variable length Command Descriptor Block formats

For all commands, if there is an invalid parameter in the command descriptor block, the target terminates the command without altering the medium.

Table 56 shows the typical format of a 6-byte CDB. Table 57 shows a typical format of a 10-byte CDB. Table 58 shows the typical format of a 12-byte CDB. Table 59 shows the typical format of a 16-byte CDB. Table 60 shows the typical format for a variable length CDB.

Table 56: Typical CDB for 6-byte commands

Bit Byte	7	6	5	4	3	2	1	0
0	Operation Code [1]							
1	Reserved			(MSB)	Logical Block Address (if required) [3]			
2								
3								(LSB)
4	Transfer Length (if required) [4] Parameter List Length (if required) [5] Allocation Length (if required) [6]							
5	Control [7]							

See notes following Table 60.

Table 57: Typical CDB for 10-byte commands

Bit Byte	7	6	5	4	3	2	1	0
0	Operation Code [1]							
1	Reserved			Service Action (if required) [2]				
2	(MSB)	Logical Block Address (if required) [3]						
3								
4								
5								(LSB)
6	Reserved							
7	(MSB)	Transfer Length (if required) [4] Parameter List Length (if required) [5] Allocation Length (if required [6]						
8								(LSB)
9	Control [7]							

See notes following Table 60.

Table 58: Typical CDB for 12-byte commands

Bit Byte	7	6	5	4	3	2	1	0
0	Operation Code [1]							
1	Reserved			Service Action (if required) [2]				
2	(MSB)	Logical Block Address (if required) [3]						
3								
4								
5								(LSB)
6	(MSB)	Transfer Length (If required) [4] Parameter List Length (if required) [5] Allocation Length (if required) [6]						
7								
8								
9								(LSB)
10	Reserved							
11	Control [7]							

See notes following Table 60.

Table 59: Typical CDB for 16-byte commands

Bit Byte	7	6	5	4	3	2	1	0
0	Operation Code [1]							
1	Reserved			Service Action (if required) [2]				
2	(MSB)	Logical Block Address (if required) [3]						
3								
4								
5								(LSB)
6	(MSB)	Additional CDB data (if required) [9]						
7								
8								
9								(LSB)
10	(MSB)	Transfer Length (If required) [4] Parameter List Length (if required) [5] Allocation Length (if required) [6]						
11								
12								
13								(LSB)
14	Reserved							
15	Control [7]							

See notes following Table 60.

Table 60: Typical variable length CDB

Bit Byte	7	6	5	4	3	2	1	0
0	Operation Code (7Fh) [1]							
1	Control [7]							
2	Reserved							
3	Reserved							
4	Reserved							
5	Encryption Identification [8]							
6	Reserved							
7	Additional CDB Length (n–7) [9]							
8	(MSB)	Service Action [2]						
9								(LSB)
10 : n	Service Action specific fields [10]							

Notes for Tables 56, 57, 58, 59, and 60.

[1] The Operation Code field of the Command Descriptor Block contains the code value indentifying the operation being requested by the CDB. The Operation Code provides for a possible 256 command operation codes. Details of the various commands with their operation codes are defined in Section 8.0 of this manual.

[2] Service Action. All typical CDB formats except the typical 6-byte format provide for a Service Action field containing a coded value identifying a function to be performed under the more general command function specified in the Operation Code field. While the Service Action field is defined for typical CDB formats, it is used as described in this clause only in those CDB formats that explicitly contain a Service Action field. When the specific field Service Action is not defined in a CDB format, the bits identified as the Service Action field in a typical CDB may be used for other purposes. More details appear on this in Section 8.0 where the details of each command are described.

[3] The Logical Block Address on logical units or within a partition on device volumes shall begin with block zero and be contiguous up to the last logical block on that logical unit or within that partition.

A six-byte command descriptor block contains a 21-bit Parameter List Length field. The 10-byte, the 12-byte and the 16-byte command descriptor blocks contain 32-bit Logical Block Address fields. Logical Block Address fields in additional parameter data have their length specified for each occurrence. See the specific command descriptions.

[4] The Transfer Length specifies the amount of data to be transferred, usually the number of blocks. For several commands the transfer length indicates the requested number of bytes to be sent as defined in the command description. For these commands the transfer length field may be identified by a different name. See the following descriptions and the individual command descriptions for further information.

Commands that use one byte for Transfer Length allow up to 256 blocks of data to be transferred by one command. A Transfer Length value of 1 to 255 indicates the number of blocks that shall be transferred. A value of zero indicates 256 blocks.

Commands that use two bytes for Transfer Length allow up to 65,535 blocks of data to be transferred by one command. In this case, a Transfer Length of zero indicates that no data transfer shall take place. A value of 1 to 65,535 indicates the number of blocks that shall be transferred.

For several commands more than two bytes are allocated for Transfer Length. Refer to the specific command description for further information.

The Transfer Length of the commands that are used to send a list of parameters to a drive is called the Parameter List Length. The Parameter List Length specifies the number of bytes sent during the DATA OUT phase.

The Transfer Length of the commands used to return sense data (e.g., REQUEST SENSE, INQUIRY, MODE SENSE, etc.) to a SCSI initiator port is called the Allocation Length. The Allocation Length specifies the number of bytes that the initiator has allocated for returned data. The drive shall terminate the DATA IN phase when Allocation Length bytes have been transferred or when all available data have been transferred to the initiator, whichever is less.

[5] The Parameter List Length field is used to specify the number of bytes sent from the Data-Out Buffer. This field is typically used in command descriptor blocks for parameters that are sent to a target (e.g., mode parameters, diagnostic parameters, log parameters, etc.). A parameter length of zero indicates that no data shall be transferred. This condition shall not be considered as an error.

[6] The Allocation Length field specifies the maximum number of bytes that an initiator has allocated for returned data. An allocation length of zero indicates that no data shall be transferred. This condition shall not be considered as an error. The target shall terminate transfers to the Data-In Buffer when allocation length bytes have been transferred or when all available data have been transferred, whichever is less. The allocation length is used to limit the maximum amount of data (e.g., sense data, mode data, log data, diagnostic data, etc.) returned to an initiator. If the information being transferred to the Data-In Buffer includes fields containing counts of the number of bytes in some or all of the data, the contents of these fields shall not be altered to reflect the truncation, if any, that results from an insufficient allocation length value, unless the standard that describes the Data-In Buffer format specifically states otherwise. If the amount of information to be transferred exceeds the maximum value that may be specified in the Allocation Length field, the target shall transfer no data and return a CHECK CONDITION status; the sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to Invalid Field in CDB.

[7] Only the Control fields have consistently defined meanings across all commands. The Control field is the last byte of every command descriptor block, except for the variable length CDB format. The Control field is defined in Table 1.

Table 1: Control field

Bit **Byte**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Last	Vendor-Specific		Reserved			NACA [a]	Obsolete [c]	Link [b] 0 or 1

[a] The Normal Auto Contingent Allegiance (NACA) bit is used to control the rules for handling an Auto Contingent Allegiance (ACA) condition caused by the command. Section 7.6.1.1 specifies the actions to be taken by the logical unit in response to an ACA condition for NACA bit values of one or zero. All drives implement support for the NACA value of zero and may support the NACA value of one. The ability to support an NACA value of one is indicated in standard INQUIRY data. See Table 77.

If the NACA bit is set to a value that is not supported, the drive completes the command with a status of Check Condition and a sense key of ILLEGAL REQUEST. The procedure for handling the resulting ACA condition operates in accordance with the supported bit value.

[b] The Link bit allows the drive to continue the task across multiple commands. Support for the Link bit is a logical unit option. A Link bit of one indicates that the initiator requests continuation of the task across two or more SCSI commands. If the Link bit is one and if the command completes successfully, a drive that supports the Link bit continues the task and returns a status of Intermediate or Intermediate–Condition Met and a service response of LINKED COMMAND COMPLETE message (see Section 4.3.6). The drive completes the command with a status of Check Condition and a sense key of ILLEGAL REQUEST if the Link bit is set to one and the drive does not support linked commands.

[c] Obsolete bit. Bit 0 provides an obsolete way to request interrupts between linked commands. If bit 0 is equal to one, targets not implementing the obsolete capability (Link bit was formerly in this bit position and Flag bit was in bit 1 position) terminate the command with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST.

[8] The Encryption Identification field indicates whether CDB bytes 8 through n and/or the data bytes are encrypted. The value also indicates which encryption key to use for decryption. A value of zero indicates no encryption. All other values are reserved.

[9] The Additional CDB Length field indicates the number of additional CDB bytes. This value in the Additional CDB Length field shall be a multiple of 4.

[10] Service Action specific fields. The Service Action field indicates the action being requested by the initiator. The Service Action field is required in the variable length CDB format and is described in Note [2]. Each service action code description defines a number of service action specific fields that are needed for that service action. If the target detects an error during decryption of encrypted CDB bytes, it shall return CHECK CONDITION status with an additional sense code of CDB Decryption Error. If the target detects an error during decryption of encrypted data bytes, it shall return CHECK CONDITION status with an additional sense code of Data Decryption Error.

7.3 Status

A Status byte shall be sent from the target to the initiator during the STATUS phase at the termination of each command as specified in Tables 61 and 62 unless the command is cleared by one of the following conditions:

1. an Abort message
2. a Bus Device Reset message
3. a hard reset condition
4. an unexpected Bus Free condition (see Section 3.1.1)
5. an ABORT TASK message
6. a CLEAR TASK SET message

Table 61: Status byte

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved		Status Byte Code					Reserved

Table 62: Status byte code bit values

Bits of status byte								Status represented
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	Good
0	0	0	0	0	0	1	0	Check Condition
0	0	0	0	0	1	0	0	Condition Met/Good
0	0	0	0	1	0	0	0	Busy
0	0	0	1	0	0	0	0	Intermediate/Good
0	0	0	1	0	1	0	0	Intermediate/Condition Met
0	0	0	1	1	0	0	0	Reservation Conflict
0	0	1	0	0	0	1	0	Obsolete [2]
0	0	1	0	1	0	0	0	Queue Full/Task Set Full [1]
0	0	1	1	0	0	0	0	ACA Active
All other codes								Reserved

[1] What was formerly called a “Command Queue” is now called a “Task Set.”

[2] Formerly “Command Terminated.”

A description of the status byte codes is given below.

Good. This status indicates that the target has successfully completed the task.

Check Condition. This status indicates that an auto contingent allegiance or contingent allegiance condition has occurred (see Section 7.6.1). Optionally, autosense data may be delivered (see Section 7.6.4.2).

Condition Met. This status shall be returned whenever the requested operation specified by an unlinked command is satisfied (see the PREFETCH commands in ANSI SCSI Block Commands-2, T10/1417D).

Busy. This status indicates that the logical unit is busy. This status shall be returned whenever a logical unit is unable to accept a command from an otherwise acceptable initiator (i.e., no reservation conflicts). The recommended initiator recovery action is to issue the command again at a later time.

Intermediate. This status or Intermediate-Condition Met shall be returned for each successfully completed command in a series of linked commands (except the last command), unless the command is terminated with Check Condition, Reservation Conflict, Task Set Full, Busy status. If Intermediate or Intermediate-Condition Met status is not returned, the series of linked commands is terminated and the task is ended.

Intermediate–Condition Met. This status is returned whenever the operation requested by a linked command is satisfied (see the PREFETCH commands in ANSI SCSI Block Commands-2, T10/1417D), unless the command is terminated with Check Condition, Reservation Conflict, Task Set Full, Busy status. If Intermediate or Intermediate-Condition Met status is not returned, the series of linked commands is terminated and the task is ended.

Reservation Conflict. This status shall be returned whenever a SCSI initiator port attempts to access a logical unit or an element of a logical unit that is reserved with a conflicting reservation type for another SCSI initiator. (See the RESERVE, RELEASE, PERSISTENT RESERVE OUT and PERSISTENT RESERVE IN commands in ANSI SCSI Primary Commands-3, T10/1416D). The recommended initiator recovery action is to issue the command again at a later time. Removing a persistent reservation belonging to a failing initiator may require the execution of a PERSISTENT RESERVE OUT command with the Preempt or Preempt and Clear actions (see the SPC-3 standard, T10/1416D).

Task Set Full. This status shall be implemented if the logical unit supports the creation of tagged tasks (see Section 7.7). This status shall not be implemented if the logical unit does not support the creation of tagged tasks.

When the logical unit has at least one task in the task set for a SCSI initiator port and a lack of task set resources prevents entering a newly received tagged task from that initiator in the task set, Task Set Full shall be returned. When the logical unit has no task in the task set for a SCSI initiator port and a lack of task set resources prevents entering a newly received tagged task from that initiator in the task set, Busy should be returned.

When the logical unit has at least one task in the task set and a lack of task set resources prevents entering a newly received untagged task in the task, Busy should be returned.

The logical unit should allow at least one queued command for each supported initiator that has identified itself to the target by a protocol specific procedure or by the successful transmission of a command.

ACA Active. This status shall be returned when an auto contingent allegiance exists within a task set and a SCSI initiator port issues a command for that task set when at least one of the following is true:

a. There is a task with the ACA attribute in the task set;
b. The initiator issuing the command did not cause the ACA condition; or
c. The task created to execute the command did not have the ACA attribute and the NACA bit was set to one in the CDB Control byte of the faulting command (see Section 7.6.1).

The initiator may reissue the command after the ACA condition has been cleared.

7.3.1 Status precedence

If more than one condition applies to a completed task, the report of a Busy, Reservation Conflict, ACA Active or Task Set Full status shall take precedence over the return of any other status for that task.

7.4 Command examples

7.4.1 Single command example

A typical operation on the SCSI bus is likely to include a single READ command to a peripheral device such as the drive. This operation is described in detail starting with a request from the initiator. This example assumes that no linked commands and no malfunctions or errors occur and is illustrated in Figure 19.

The initiator has active pointers and a set of stored pointers representing active disconnected SCSI devices (a SCSI initiator port without disconnect capability does not require stored pointers). The initiator sets up the active pointers for the operation requested, arbitrates for the SCSI bus, and selects the drive. Once this process is completed, the drive assumes control of the operation.

The drive obtains the command from the initiator (in this case a READ command). The drive interprets the command and executes it. For this command, the drive reads the requested data from the Disc Media and sends this data to the initiator. After sending the read data to the initiator, the drive sends a status byte to the initiator. To end the operation, the drive sends a Command Complete message to the initiator and then goes to the Bus Free state.

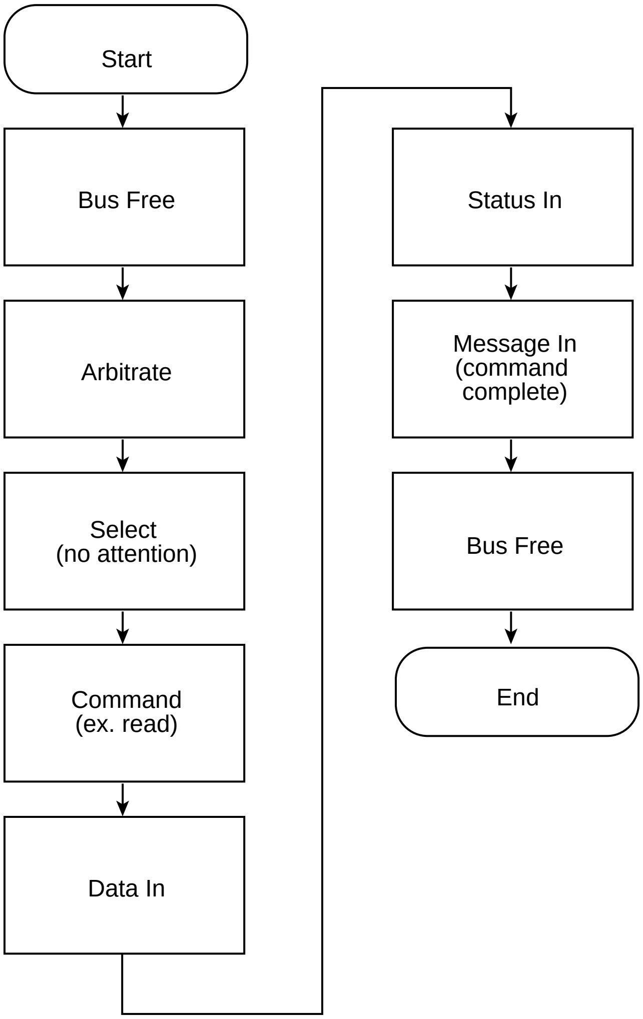


Figure 19. Single command example

7.4.2 Disconnect example

In the single command example, the length of time necessary to obtain the data may require a time consuming physical seek. In order to improve system throughput, the drive may disconnect from the initiator, freeing the SCSI bus to allow other requests to be sent to other SCSI devices. To do this, the initiator must be reselectable and capable of restoring the pointers upon reconnection. The drive must be capable of arbitrating for the SCSI bus and reselecting the initiator. See Figure 20.

After the drive has received the READ command (and has determined that there will be a delay), it disconnects by sending a DISCONNECT message and releasing BSY (goes to Bus Free state).

When the data is ready to be transferred, the drive reconnects to the initiator, the initiator restores the pointers to their most recently saved values (which in this case are the initial values), and the drive continues (as in the single command example) to finish the operation. The initiator recognizes that the operation is complete when a Command Complete message is received.

If the drive elects to disconnect after transferring part of the data (e.g., while crossing a cylinder boundary), it sends a Save Data Pointer message and a DISCONNECT message to the initiator and then disconnects. When reconnection is completed, the initiator restores the current data pointer to the value it was immediately before the Save Data Pointer message.

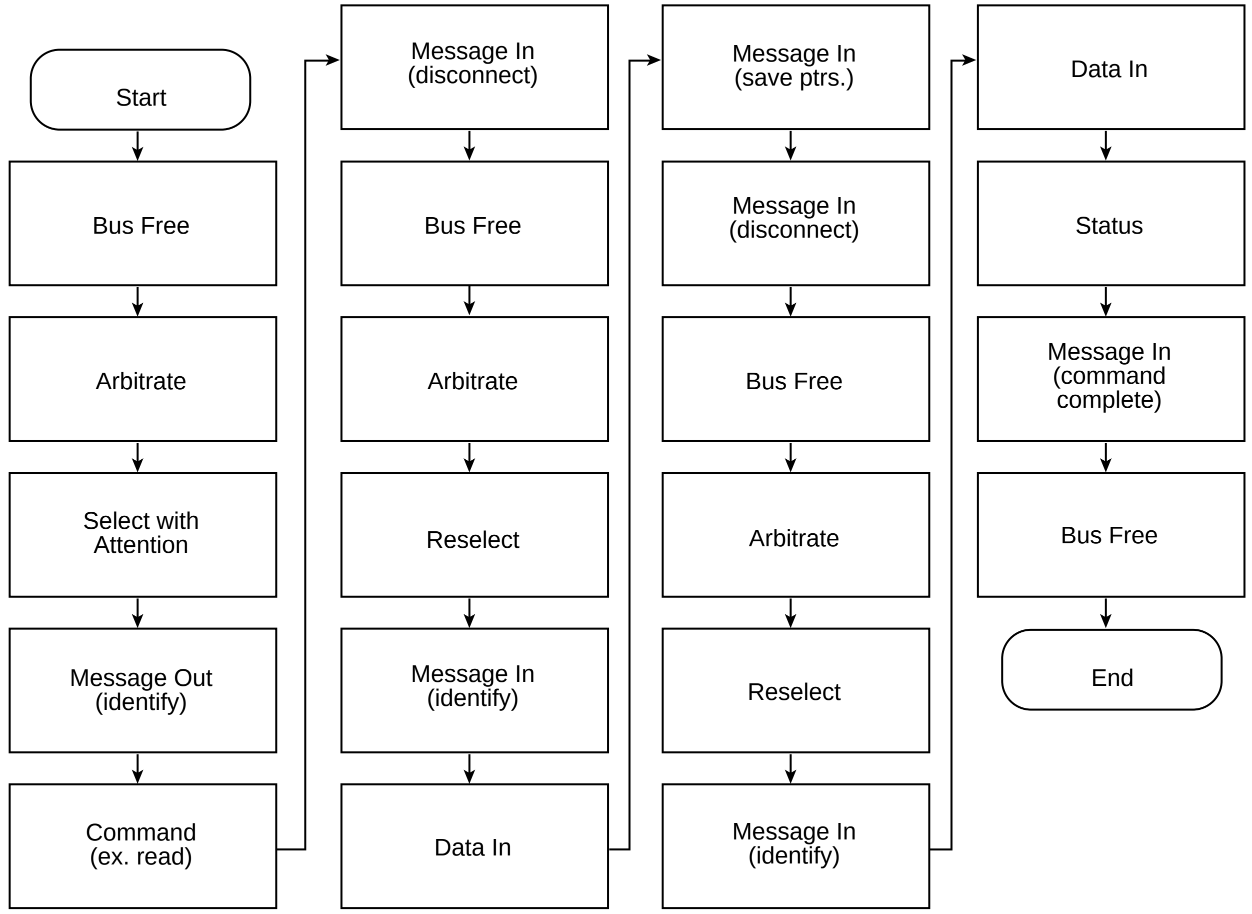


Figure 20. Disconnect example

7.5 Timing examples

Times (T00 through T35) necessary to define performance are listed in the individual drive's Product Manual, Volume 1, in the "Disc drive SCSI timing" section. For timing waveforms to define these times, refer to the SCSI Parallel Interface-4 specification (SPI-4).

7.6 Command processing considerations and exception conditions

The following clauses describe some exception conditions and errors associated with command processing and the sequencing of commands.

7.6.1 Auto Contingent Allegiance or Contingent Allegiance

The auto contingent allegiance (NACA=1, see Section 7.2.1, note [7], Table 1) or contingent allegiance (NACA=0) condition shall exist within the task set when the logical unit completes a command by returning a CHECK CONDITION status (see Section 7.3).

7.6.1.1 Logical Unit response to Auto Contingent Allegiance or Contingent Allegiance

The auto contingent allegiance (NACA=1, see Section 7.2.1, note [7], Table 1) or contingent allegiance (NACA=0) condition shall not cross task set boundaries and shall be preserved until it is cleared as described in Section 7.6.1.2. If requested by the initiator and supported by the protocol and logical unit, sense data shall be returned as described in Section 7.6.4.2.

Note. The SCSI-2 contingent allegiance condition has had an alternate added and the extended contingent allegiance condition has been replaced in SCSI-3 (now called merely "SCSI") by auto contingent allegiance in conjunction with the NACA bit.

Note. If the SCSI-3 protocol does not enforce synchronization of client/server states, there may be a time delay between the occurrence of the auto contingent allegiance or contingent allegiance condition and the time at which the initiator becomes aware of the condition.

After sending status and a service response of Task Complete, the logical unit shall modify the state of all tasks in the faulted task set as described in SCSI Architecture Model-2 (SAM-2), T10/1157D, clause 7, "Task Set Management."

A task created by the faulted initiator while the auto contingent allegiance condition is in effect may be entered into the faulted task set under the conditions described below.

As described in Section 7.2.1, note [7], Table 1, the setting of the NACA bit in the Control byte of the faulting command CDB determines the rules that apply to an ACA or CA condition caused by that command. If the NACA bit was set to zero the SCSI-2 contingent allegiance rules shall apply.

If the NACA bit was set to one in the Control byte of the faulting command, then a new task created by the faulted initiator while the ACA condition is in effect shall not be entered into the faulted task set unless all of the following conditions are true:

a. The task has the ACA attribute; and
b. No other task from the faulted initiator having the ACA attribute is in the task set.

If the task is from the faulted initiator and any of the conditions listed above are not met, the newly created task shall not be entered into the task set and shall be completed with a status of ACA Active.

If a task having the ACA attribute is received and no auto contingent allegiance condition is in effect for the task set or if the NACA bit was set to zero in the CDB for the faulting command (i.e., a contingent allegiance condition is in effect), then the ACA task shall be completed with a status of Check Condition. The sense key shall be set to ILLEGAL REQUEST with an additional sense code of Invalid Message Error. As noted in Section 7.2.1, note [7], Table 1, any existing contingent allegiance condition shall be cleared and a new auto contingent allegiance (NACA=1) or contingent allegiance (NACA=0) condition shall be established.

The handling of tasks created by initiators other than the faulted initiator depends on the value in the TST field in the Control mode page (see ANSI SCSI Primary Commands-3, T10/1416D).

If TST=000b, tasks created by other initiators while the ACA or CA condition is in effect shall not be entered into the faulted task set (except for a PERSISTENT RESERVE command with a Preempt and Clear action as described in Section 7.2.1, note [7], Table 1). Tasks rejected from the task set due to the presence of an ACA or CA condition shall be completed with a status of ACA Active (if NACA=1 in the new command's CDB Control byte, see Section 7.2.1, note [7], Table 1) or Busy (if NACA=0).

If TST=001b, tasks created by one initiator shall not be rejected based on an ACA or CA condition in effect for another initiator. Only ACA or CA condition for the sending initiator (as well as other task set management considerations described in SCSI Architecture Model-2, T10/1157D, clause 7, "Task Set Management.") shall affect acceptance into the task set or rejection for a task from that initiator.

7.6.1.2 Clearing an Auto Contingent Allegiance condition

If the NACA bit is set to zero in the Control byte of the faulting command, then the SCSI-2 rules for clearing contingent allegiance shall apply. In addition, the logical unit shall clear the associated contingent allegiance condition upon sending sense data by means of the autosense mechanism described in Section 7.6.4.2.

While the SCSI rules for clearing the CA condition are in effect, a logical unit that supports the Clear ACA task management function shall ignore all Clear ACA requests and shall return a service response of Function Complete (see ANSI SCSI Architecture Model-2, T10/1157D, Section 6.3).

If the logical unit accepts a value of one for the NACA bit and this bit was set to one in the Control byte of the faulting command, then the SCSI-2 rules for clearing a contingent allegiance condition shall not apply. In this case, the ACA condition shall only be cleared:

a. As the result of a power on or a logical unit reset (see ANSI SAM-2 document T10/1157D, Section 5.6.1.2);

b. Through a Clear ACA task management function issued by the faulting initiator as described in ANSI SAM-2 document T10/1157D;

c. Through a Preempt and Clear action of a PERSISTENT RESERVE OUT command that clears the tasks of the faulting initiator (see the ANSI SCSI Primary Commands-2, T10/1236D);

d. A command with the ACA attribute terminates with a CHECK CONDITION status. The state of all tasks in the task set when an auto contingent allegiance condition is cleared shall be modified as described in ANSI document T10/1157D, clause 7, "Task Set Management."

7.6.2 Overlapped commands

An overlapped command occurs when an initiator reuses a Task Address (see Glossary) in a new command before a previous task to which that address was assigned completes its task lifetime as described in SCSI Architecture Model-2 (SAM-2), T10/1157D, clause 7, "Task and Command Lifetimes."

Each system that implements a SCSI protocol standard shall specify whether or not a logical unit is required to detect overlapped commands. A logical unit that detects an overlapped command shall abort all tasks for the initiator in the task set and shall return CHECK CONDITION status for that command. If the overlapped command condition was caused by an untagged task or a tagged task with a tag value exceeding FFh, then the sense key shall be set to Aborted Command and the additional sense code shall be set to Overlapped Commands Attempted. Otherwise, an additional sense code of Tagged Overlapped Tasks shall be returned with the additional sense code qualifier byte set to the value of the duplicate tag.

Note. An overlapped command may be indicative of a serious error and, if not detected, could result in corrupted data. 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 logical unit 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).

Note. Some logical units may not detect an overlapped command until after the command descriptor block has been received.

7.6.3 Incorrect logical unit selection

The target's response to an incorrect logical unit identifier is described in the following paragraphs.

The logical unit identifier may be incorrect because:

a. The target does not support the logical unit (e.g., some targets support only one peripheral device). In response to any other command except REQUEST SENSE and INQUIRY, the target shall terminate the command with CHECK CONDITION status. Sense data shall be set to the values specified for the REQUEST SENSE command in item b below;

b. The target supports the logical unit, but the peripheral device is not currently attached to the target. In response to an INQUIRY command the target shall return the INQUIRY data with the peripheral qualifier set to the value required in the SPC-3 standard. In response to a REQUEST SENSE command, the target shall return sense data. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to Logical Unit Not Supported.

   In response to any other command except REQUEST SENSE and INQUIRY, the target shall terminate the command with CHECK CONDITION status. Sense data shall be set to the values specified for the REQUEST SENSE command above;

c. The target supports the logical unit and the peripheral device is attached, but not operational. In response to an INQUIRY command the target shall return the INQUIRY data with the peripheral qualifier set to the value required in Section 8.0. In response to REQUEST SENSE, the target shall return sense data.

   The target's response to any command other than INQUIRY and REQUEST SENSE is vendor-specific; or

d. The target supports the logical unit but is incapable of determining if the peripheral device is attached or is not operational when it is not ready.

   In response to an INQUIRY command the target shall return the INQUIRY data with the peripheral qualifier set to the value specified in Section 8.0. In response to a REQUEST SENSE command the target shall return the REQUEST SENSE data with a sense key of No Sense unless an auto contingent allegiance exists. The target's response to any other command is vendor-specific.

7.6.4 Sense data

Sense data shall be made available by the logical unit in the event a command completes with a CHECK CONDITION status or other conditions. The format, content and conditions under which sense data shall be prepared by the logical unit are specified in this manual, the SPC-3 standard, and other applicable SCSI protocol standards.

Sense data shall be preserved by the logical unit for the initiator until it is transferred by one of the methods listed below or until another task from that initiator is entered into the task set.

The sense data may be transferred to the initiator through any of the following methods:

a. The REQUEST SENSE command specified in Section 8.38;
b. An asynchronous event report; or
c. Autosense delivery.

The following clauses describe the last two transfer methods.

7.6.4.1 Asynchronous Event Reporting

Asynchronous Event Reporting is used by a logical unit to signal another device that an asynchronous event has occurred. The mechanism automatically returns sense data associated with the event. Support for asynchronous event reporting is a logical unit option.

The control mode page contains parameters affecting the use of asynchronous event reporting (see Section 8.13.9).

Asynchronous Event Reporting is used to signal a device that one of the four events listed below has occurred:

a. an error condition was encountered after command completion;
b. a newly initialized device is available;
c. some other type of unit attention condition has occurred; or
d. an asynchronous event has occurred.

An example of the first case above occurs in a device that implements a write cache. If the target is unable to write cache data to the medium, it may use an asynchronous event report to inform the initiator of the failure.

An example of the second case above is a logical unit that generates an asynchronous event report, following a power-on cycle, to notify other SCSI devices that it is ready to accept I/O commands.

Sense data accompanying the report identifies the condition (see Section 7.6.4).

An error condition or unit attention condition shall be reported to a specific initiator once per occurrence of the event causing it. The logical unit may choose to use an asynchronous event report or to return CHECK CONDITION status on a subsequent command, but not both. Notification of an error condition encountered after command completion shall be returned only to the initiator that sent the affected task or tasks.

Asynchronous event reports may be used to notify devices that a system resource has become available. If a logical unit uses this method of reporting, the sense key in the AER sense data shall be set to Unit Attention.

7.6.4.2 Autosense

Autosense is the automatic return of sense data to the initiator coincident with the completion of an SCSI command under the conditions described below. The return of sense data in this way is equivalent to an explicit command from the initiator requesting sense data immediately after being notified that an ACA condition has occurred. Inclusion of autosense support in an SCSI protocol standard is optional.

An initiator may request autosense service for any SCSI command. If supported by the protocol and logical unit and requested by the initiator, the target shall only return sense data in this manner coincident with the completion of a command with a status of Check Condition. After autosense data is sent, the sense data and the CA (NACA=0), if any, shall then be cleared. Autosense shall not affect ACA (NACA=1), see Section 7.6.1.

Protocol standards that support autosense shall require an autosense implementation to:

a. Notify the logical unit when autosense data has been requested for a command; and

b. Inform the initiator when autosense data has been returned upon command completion (see Section 7.0).

It is not an error for the initiator to request the automatic return of sense data when autosense is not supported by the SCSI protocol or logical unit implementation. If the initiator requested the return of sense data through the autosense facility and the protocol service layer does not support this feature, then the confirmation returned by the initiator's service delivery port should indicate that no sense data was returned. If the protocol service layer supports autosense but the logical unit does not, then the target should indicate that no sense data was returned. In either case, sense information shall be preserved and the initiator may issue a command to retrieve it.

7.6.5 Unexpected RESELECTION phase

An unexpected RESELECTION phase occurs if a SCSI target port attempts to do a physical reconnect to a task for which a nexus does not exist. a SCSI initiator port should respond to an unexpected RESELECTION phase by sending an ABORT TASK message.

7.6.6 Unit Attention condition

The drive sets up the Unit Attention condition when it stores (within itself) a Unit Attention condition flag for each device on the SCSI bus having a SCSI initiator port relationship with the drive, and this Unit Attention condition persists for each initiator until the condition is cleared (flag negated) by each initiator individually. The Unit Attention condition results when one of the following events occur:

a. A power-on sequence occurs.

b. A reset is generated internally by the drive (caused by a power glitch).

c. A Bus Device Reset message causes the drive to reset itself.

d. The Reset I/O line resets the drive.

e. a SCSI initiator port changes one or more of the mode select parameters in the drive (these changes could affect one or more of the other initiators).

f. The inquiry data has been changed.

g. The mode parameters in effect for a SCSI initiator port have been restored from non-volatile memory.

h. An event occurs that requires the attention of the initiator.

i. A Clear Queue message is received.

j. The Log parameters are changed. Unit Attention condition is posted for all initiators in the system other than the one that changed the Log Parameters.

k. Tasks for this initiator were changed by another initiator.

l. The version or level of microcode has changed.

m. The logical unit inventory has been changed (not generally applicable to Seagate disc drives); or

n. A change in the condition of a synchronized spindle occurred (not generally applicable to Seagate disc drives).

The Unit Attention Parameters page (page 00h, bit 4 of byte 2) of the MODE SELECT command controls whether or not a CHECK CONDITION status is to be reported to affected initiators when a Unit Attention condition exists (see Table 155).

Logical units may queue unit attention conditions. After the first unit attention condition is cleared, another unit attention condition may exist (for example, a power on condition followed by a microcode change condition).

The Unit Attention condition for a particular initiator is cleared when that initiator does one of the following:

a. It sends a REQUEST SENSE command; or

b. It sends any other legitimate command, with the exception of the INQUIRY command. The INQUIRY command does not clear the Unit Attention condition.

When a Unit Attention condition flag is stored in the drive for a SCSI initiator port, the commands that initiator issues to the drive operate as described in the following paragraphs.

If a SCSI initiator port sends an INQUIRY command to the drive when the drive has stored a Unit Attention condition flag for that initiator (before the drive generates the auto contingent allegiance condition), the drive shall perform the INQUIRY command and shall not clear the Unit Attention condition.

If a SCSI initiator port sends a REQUEST SENSE command to the drive when a Unit Attention condition flag is stored for that initiator (before the drive generates the auto contingent allegiance condition), the drive shall discard any pending sense data, report the Unit Attention sense key, and clear the Unit Attention condition (negate the flag) for that initiator.

If a SCSI initiator port issues a command other than INQUIRY or REQUEST SENSE while a Unit Attention condition flag is stored for that initiator, the drive may or may not perform the command and report CHECK CONDITION status, depending on whether or not the Unit Attention bit is zero or one in the Unit Attention Mode Parameters page (Page 00h, bit 4 of byte 2). See Table 155. If a REQUEST SENSE is issued next, the Unit Attention condition is reported and cleared (flag negated) as noted in the preceding paragraph. If another command other than REQUEST SENSE or INQUIRY is issued instead, the drive shall perform the command and return the appropriate status. The Unit Attention condition for the subject initiator is cleared (flag negated) and the sense data and flag indicating there has been a Unit Attention condition are lost.

7.6.7 Target hard reset

a SCSI target port hard reset is a SCSI target port response to a SCSI target port Reset task management request (see SCSI Architecture Model-2, T10/1157D), or a reset event within the service delivery subsystem. The definition of target reset events is protocol and interconnect specific. Each SCSI product standard shall specify the response to a SCSI target port reset event including the conditions under which a SCSI target port hard reset shall be executed.

To execute a hard reset, a SCSI target port shall initiate a logical unit reset for all attached logical units as described in Section 7.6.8.

7.6.8 Logical unit reset

A logical unit reset is a response to a Logical Unit Reset task management request (see SCSI Architecture Model-2, T10/1157D), or some other logical unit reset event, such as a SCSI target port hard reset (see Section 7.6.7). The definition of such events may be device-specific or dependent on the protocol and interconnect. Each appropriate SCSI standard shall specify the conditions under which a logical unit reset shall be executed.

To execute a logical unit reset the logical unit shall:

1. Abort all tasks in its task set;
2. Clear an auto contingent allegiance condition, if one is present;
3. Release all SCSI device reservations;
4. Return the device’s operating mode to the appropriate initial conditions, similar to those conditions that would be found following device power-on. The mode select conditions (see SCSI Primary Commands-2 SPC-3, T10/1416D) shall be restored to their saved values if saved values have been established. Mode select conditions for which no saved values have been established shall be returned to their default values;
5. Set a Unit Attention condition (see Section 7.6.6); and
6. Initiate a logical unit reset for all nested logical units.

In addition to the above, the logical unit shall execute any additional functions required by the applicable standards.

7.7 Queued tasks (formerly “queued I/O processes”)

Queuing of tasks allows a drive to accept multiple commands for execution at a later time.

There are two methods for implementation of queuing: tagged and untagged. Tagged task queuing allows the drive to accept multiple commands from each initiator. Untagged task queuing allows the drive to accept one command from each initiator. Drives that have SCSI-2 implementation support tagged queuing while in SCSI-1 or SCSI-2 mode. They can use untagged task queuing mode if the initiator does not send task queue tag messages.

Initiators may add or delete commands to the queue for the drive within the limitations specified in this specification. When adding a command, the initiator may specify fixed order of execution, allow the drive to define the order of execution, or specify that the command is to be executed next. See glossary in Section 1.2.1 for terminology definitions when reading the following explanations.

7.7.1 Untagged task queuing

Untagged task queuing allows the drive to accept a command from a SCSI initiator port while a command from another initiator is being executed. Only one command for each I T L nexus may be accepted at a time.

A new task may be initiated any time the BUS FREE phase exists even if another task from a different initiator is being executed. If the disconnect privilege is not granted, the drive returns Busy status to the new task.

The I T L nexus specifies the relationship so that the drive can always reconnect to the initiator to restore the pointers for task as long as only one command per I T L nexus is issued. It is the responsibility of the initiator to assure that only one command is issued at any time.

7.7.2 Tagged task queuing

Tagged task queuing allows a drive to accept multiple commands from the same or different initiators until the drive’s task queue is full. A new task may be initiated any time the BUS FREE phase exists, if the disconnect privilege is granted. If the disconnect privilege is not granted for a tagged command, the drive returns Busy status to the new task.

The Task Queue Tag messages (see Section 4.4) allow the initiator to establish a unique I T L Q nexus to identify each task. Each task may be a command or a set of linked commands with a unique queue tag.

The I T L Q nexus allows the target to reconnect to the desired task and the initiator to restore the correct set of pointers. a SCSI initiator port may have several task ongoing to the same or different logical unit as long as each has a unique nexus.

If only Simple Task Queue Tag messages are used, the drive may execute the commands in any order that is deemed desirable within the constraints of the queue management algorithm specified in the Control Mode page (see Table 162). The command ordering is done by the drive to meet its performance and functional goals. The algorithm used by the drive attempts to achieve certain drive or system performance goals established in the drive firmware for the queued commands and guarantee that all commands will be executed. One possible goal would be to minimize seek times, but there could be others, possibly designed to meet some special system need. Commands from other initiators are also executed in an order selected in the same manner. The drive uses the Simple Task Queue Tag when reconnecting to the initiator.

If Ordered Task Queue Tag messages are used, the drive executes the commands in the order received with respect to other commands received with Ordered Task Queue Tag messages. All commands received with a Simple Task Queue Tag message prior to a command received with an Ordered Task Queue Tag message, regardless of initiator, are executed before that command with the Ordered Task Queue Tag message. All commands received with a Simple Task Queue Tag message after a command received with an Ordered Task Queue Tag message, regardless of initiator, are executed after that command with the Ordered Task Queue Tag message.

A command received with a Head of Task Queue Tag message is placed first in the queue, to be executed next. A command received with a Head of Task Queue Tag message does not suspend a task for which the drive has begun execution. Consecutive commands received with Head of Task Queue Tag messages are executed in a last-in-first-out order.

The Control Mode page specifies alternative queue management algorithms with additional rules on the order of execution of commands (see Table 162).

A task received from a SCSI initiator port without a task queue tag message while there are any tagged I/O commands in the command queue from that initiator is an incorrect initiator connection, unless there is a contingent allegiance condition. A task received from a SCSI initiator port with a task queue tag message while there is an untagged command in the command queue from that initiator is also an incorrect initiator connection. In either of these cases, the drive removes all commands in the queue from that initiator, aborts the command in process if it is from that initiator, and sets the Sense Key to Aborted Command and the Sense Code to Overlapped Commands Attempted.

The RESERVE and RELEASE commands should be sent with an Ordered Task Queue Tag message. Use of the Head of Task Queue Tag message with these commands could result in reservation conflicts with previously issued commands.

The TEST UNIT READY and INQUIRY commands are often sent with a Head of Task Queue Tag message, since the information returned is either available or has no effect on the condition of the drive.

The drive recovery option, is to continue execution of commands in the queue after the contingent allegiance condition has cleared. The drive returns Busy status to all other initiators while the contingent allegiance condition exists. During this time all commands in the queue are suspended. All commands used for recovery operations are untagged commands.

Deferred errors are normally related to a command that has already completed. As such, there is no attempt to return the queue tag value assigned to the original command.

7.8 Parameter rounding

Certain parameters sent to a SCSI target port with various commands contain a range of values. Targets may choose to implement only selected values from this range. When the target receives a value that it does not support, it either rejects the command (CHECK CONDITION status with Illegal Request Sense key) or it rounds the value received to a supported value. The target shall reject unsupported values unless rounding is permitted in the description of the parameter.

Rounding of parameter values, when permitted (Rounding is enabled by MODE SELECT command, page code 00h, byte 2, bit 2) shall be performed as follows:

a SCSI target port that receives a parameter value that is not an exact supported value shall adjust the value to one that it supports and shall return CHECK CONDITION status with a sense key of Recovered Error. The additional sense code shall be set to Rounded Parameter. The initiator is responsible to issue an appropriate command to learn what value the target has selected.

Implementors Note: Generally, the target should adjust maximum-value fields down to the next lower supported value than the one specified by the initiator. Minimum-value fields should be rounded up to the next higher supported value than the one specified by the initiator. In some cases, the type of rounding (up or down) is explicitly specified in the description of the parameter.

7.9 Programmable operating definition

Some applications require that the operating definition of a logical unit be modified to meet the special requirements of a particular initiator. The program-controlled modification of the operating definition is provided to allow operating systems to change the operating definition of a more recently developed target to one which is more compatible with the operating system. This ability requires that the system comply with the low-level hardware definitions of SCSI-2.

The parameters that can be changed by modifying the operating definition of a logical unit include the vendor identification, the device type, the device model, the SCSI compliance level, the SCSI specification level, the command set, and other parameters. The low-level hardware parameters including signal timing and parity definitions cannot be changed by modifying the operating definition. The present operating definition of a logical unit with respect to a SCSI initiator port can be determined at any time by execution of an INQUIRY command. In some vendor-specific cases, it may also be necessary to perform other commands including MODE SENSE and READ CAPACITY.

The more recent Seagate drives do not support the Change Operating Definition command (see individual drive’s Product Manual, Volume 1).

Each logical unit begins at a particular operating definition. If the logical unit supports the CHANGE DEFINITION command, the present operating definition can be changed to any other operating definition supported by the logical unit. The actual details of the operating definition of a logical unit are vendor-specific. If the operating definition is changed to one that does not include the CHANGE DEFINITION command, the target continues to accept the CHANGE DEFINITION command.

If an error occurs during execution of a CHANGE DEFINITION command, the original operating definition remains in effect after the command is executed. The new operating definition becomes active only after successful execution of the CHANGE DEFINITION command.

Since new operating definitions may preclude the execution of tasks that are already in progress, the target may disconnect to allow completion of any tasks that are in progress. Operating definition changes that may cause conflicts with the normal operation from other initiators shall be indicated to those initiators by generating a Unit Attention condition for each other initiator. The additional sense code shall be set to Changed Operating Definition.

A SCSI initiator port may request a list of the operating definitions that the target supports and descriptive text for each operating definition using the INQUIRY command.

7.10 Incorrect initiator connection

An incorrect initiator connection occurs on a reconnection if:

a. a SCSI initiator port attempts to reconnect to a task, and

b. a soft reset condition has not occurred, and

c. the initiator does not send an Abort, Abort Tag, Bus Device Reset, Clear Task Set, Continue Task, or Terminate Task message during the same MESSAGE OUT phase as the IDENTIFY message.

An incorrect initiator connection also occurs on an initial connection when a SCSI initiator port:

a. attempts to establish an I T L Q nexus when an I T L nexus already exists from a previous connection, or

b. attempts to establish an I T L nexus when an I T L Q nexus already exists unless there is a contingent allegiance or extended contingent allegiance condition present for the logical unit or target routine.

A SCSI target port that detects an incorrect initiator connection shall abort all tasks for the initiator on the logical unit or target routine and shall return CHECK CONDITION status. The sense key shall be set to Aborted Command and the additional sense code shall be set to Overlapped Commands Attempted.

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, host-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. Also, some targets may not detect an incorrect initiator connection until after the command descriptor block has been received.

8.0 Command descriptions

Two types of commands are supported by the drive: commands for all devices; and commands for direct access devices. The individual Product Manuals for each Seagate model drive list the Commands and parameter pages that the particular drive supports. Some commands are not supported by any Seagate devices. This manual states which these are when the commands are listed herein. The exception to this is for commands or features that will likely be supported in the future, either as standard items or as optional (customer request) items.

Table 63 lists commands alphabetically by command name. Table 64 lists commands by operation code.

Table 63. Commands sorted by command name

Command name	Op code [1]	Command type		Length						Reference
		All devices	Direct access devices	6 byte	10 byte	12 byte	16 byte	32 byte	Variable	
CHANGE DEFINITION	40h	x [2]								8.1 (see page 160)
COMPARE	39h	x [2]								8.2 (see page 160)
COPY	18h	x [2]								8.3 (see page 160)
COPY AND VERIFY	3Ah	x [2]								8.4 (see page 160)
FORMAT UNIT	04h		x	x						8.5 (see page 161)
INQUIRY	12h	x		x						8.6 (see page 170)
LOCK-UNLOCK CACHE (10)	36h		x		x					8.7 (see page 197)
LOCK-UNLOCK CACHE (16)	92h		x				x			8.8 (see page 198)
LOG SELECT	4Ch	x			x					8.9 (see page 200)
LOG SENSE	4Dh	x			x					8.10 (see page 203)
MODE SELECT (6)	15h		x	x						8.11 (see page 227)
MODE SELECT (10)	55h		x		x					8.12 (see page 232)
MODE SENSE (6)	1Ah		x	x						8.13 (see page 235)
MODE SENSE (10)	5Ah		x		x					8.14 (see page 281)
MOVE MEDIUM	A7h									8.15 (see page 284)
PERSISTENT RESERVE IN	5Eh		x							8.16 (see page 285)
PERSISTENT RESERVE OUT	5Fh		x							8.17 (see page 292)
PREFETCH (10)	34h		x		x					8.18 (see page 297)
PREFETCH (16)	90h		x				x			8.19 (see page 299)
PREVENT/ALLOW MEDIUM REMOVAL	1Eh		x							8.20 (see page 300)
READ (6)	08h		x	x						8.21 (see page 301)
READ (10) [3]	28h		x		x					8.22 (see page 303)
READ (12)	A8h		x			x				8.23 (see page 305)

Table 63. Commands sorted by command name (continued)

Command name	Op code [1]	Command type		Length						Reference
		All devices	Direct access devices	6 byte	10 byte	12 byte	16 byte	32 byte	Variable	
READ (16)	88h		x				x			8.24 (see page 307)
READ BUFFER	3Ch	x			x					8.25 (see page 309)
READ CAPACITY (10)	25h		x		x					8.26 (see page 313)
READ CAPACITY (16)	9Eh		x				x		10h	8.27 (see page 315)
READ DEFECT DATA (10)	37h		x		x					8.28 (see page 317)
READ DEFECT DATA (12)	B7h		x			x				8.29 (see page 319)
READ ELEMENT STATUS	B4h		x							8.30 (see page 321)
READ LONG	3Eh		x		x					8.31 (see page 322)
REASSIGN BLOCKS	07h		x	x						8.32 (see page 324)
RECEIVE DIAGNOSTIC RESULTS	1Ch	x		x						8.33 (see page 326)
RELEASE (6)	17h		x [2]	x						8.34 (see page 330)
RELEASE (10)	57h		x [2]		x					8.35 (see page 330)
REPORT DEVICE IDENTIFIER	A3h	x				x			05h	8.36 (see page 331)
REPORT LUNS	A0h		x							8.37 (see page 333)
REQUEST SENSE	03h	x		x						8.38 (see page 336)
RESERVE (6)	16h		x [2]	x						8.39 (see page 348)
RESERVE (10)	56h		x [2]		x					8.40 (see page 348)
REZERO UNIT	01h		x [2]	x						8.41 (see page 348)
SEARCH DATA EQUAL	31h		x [2]							8.42 (see page 348)
SEARCH DATA HIGH	30h		x [2]							8.43 (see page 348)
SEARCH DATA LOW	32h		x [2]							8.44 (see page 348)
SEEK (6)	0Bh		x [2]	x						8.45 (see page 348)
SEEK EXTENDED (10)	2Bh		x		x					8.46 (see page 349)
SEND DIAGNOSTIC	1Dh	x		x						8.47 (see page 350)
SET DEVICE IDENTIFIER	A4h	x				x			06h	8.48 (see page 354)
SET LIMITS	33h		x							8.49 (see page 355)
START/STOP UNIT	1Bh		x	x						8.50 (see page 356)
SYNCHRONIZE CACHE (10)	35h		x		x					8.51 (see page 357)
SYNCHRONIZE CACHE (16)	91h		x				x			8.52 (see page 358)
TEST UNIT READY	00h	x		x						8.53 (see page 360)
VERIFY(10)	2Fh		x		x					8.54 (see page 361)
VERIFY (12)	AFh		x			x				8.55 (see page 363)
VERIFY (16)	8Fh		x				x			8.56 (see page 365)
WRITE (6)	0Ah		x	x						8.57 (see page 367)
WRITE (10)	2Ah		x		x					8.58 (see page 369)
WRITE (12)	AAh		x			x				8.59 (see page 371)
WRITE (16)	8Ah		x				x			8.60 (see page 373)
WRITE AND VERIFY (10)	2Eh		x		x					8.61 (see page 375)
WRITE AND VERIFY (12)	AEh		x			x				8.62 (see page 377)
WRITE AND VERIFY (16)	8Eh		x				x			8.63 (see page 379)

Table 63. Commands sorted by command name (continued)

		Command type		Length						
Command name	Op code [1]	All devices	Direct access devices	6 byte	10 byte	12 byte	16 byte	32 byte	Variable	Reference
WRITE BUFFER	3Bh	x			x					8.64 (see page 381)
WRITE LONG	3Fh		x		x					8.65 (see page 386)
WRITE SAME (10)	41h		x		x					8.66 (see page 387)
WRITE SAME (16)	93h		x				x			8.67 (see page 388)
XDREAD (10)	52h		x							8.69 (see page 390)
XDREAD (32)	7Fh		x					x	0003h	8.70 (see page 391)
XDWRITE (10)	50h		x							8.71 (see page 393)
XDWRITE (32)	7Fh		x					x	0004h	8.72 (see page 394)
XDWRITEREAD (10)	53h		x		x					8.73 (see page 396)
XDWRITEREAD (32)	7Fh		x					x	0007h	8.74 (see page 397)
XPWRITE (10)	51h		x		x					8.75 (see page 399)
XPWRITE (32)	7Fh		x					x	0006h	8.76 (see page 400)

[1] Command operation code.

[2] Obsolete command.

[3] Formerly called READ EXTENDED.

Table 64. Commands sorted by operation code

		Command type		Length						
Op code [1]	Command name	All devices	Direct access devices	6 byte	10 byte	12 byte	16 byte	32 byte	Variable	Reference
00h	TEST UNIT READY	x		x						8.53 (see page 360)
01h	REZERO UNIT		x [2]	x						8.41 (see page 348)
03h	REQUEST SENSE	x		x						8.38 (see page 336)
04h	FORMAT UNIT		x	x						8.5 (see page 161)
07h	REASSIGN BLOCKS		x	x						8.32 (see page 324)
08h	READ (6)		x	x						8.21 (see page 301)
0Ah	WRITE (6)		x	x						8.57 (see page 367)
0Bh	SEEK (6)		x [2]	x						8.45 (see page 348)
12h	INQUIRY	x		x						8.6 (see page 170)
15h	MODE SELECT (6)		x	x						8.11 (see page 227)
16h	RESERVE (6)		x [2]	x						8.39 (see page 348)
17h	RELEASE (6)		x [2]	x						8.34 (see page 330)
18h	COPY	x [2]								8.3 (see page 160)
1Ah	MODE SENSE (6)		x	x						8.13 (see page 235)
1Bh	START/STOP UNIT		x	x						8.50 (see page 356)
1Ch	RECEIVE DIAGNOSTIC RESULTS	x		x						8.33 (see page 326)
1Dh	SEND DIAGNOSTIC	x		x						8.47 (see page 350)
1Eh	PREVENT/ALLOW MEDIUM REMOVAL		x							8.20 (see page 300)
25h	READ CAPACITY (10)		x		x					8.26 (see page 313)
28h	READ (10) [3]		x		x					8.22 (see page 303)
2Ah	WRITE (10)		x		x					8.58 (see page 369)
2Bh	SEEK EXTENDED (10)		x		x					8.46 (see page 349)
2Eh	WRITE AND VERIFY (10)		x		x					8.61 (see page 375)
2Fh	VERIFY (10)		x		x					8.56 (see page 365)
30h	SEARCH DATA HIGH		x [2]							8.43 (see page 348)
31h	SEARCH DATA EQUAL		x [2]							8.42 (see page 348)
32h	SEARCH DATA LOW		x [2]							8.44 (see page 348)
33h	SET LIMITS		x							8.49 (see page 355)
34h	PREFETCH (10)		x							8.18 (see page 297)
35h	SYNCHRONIZE CACHE (10)		x		x					8.51 (see page 357)
36h	LOCK-UNLOCK CACHE (10)		x							8.7 (see page 197)
37h	READ DEFECT DATA (10)		x		x					8.28 (see page 317)
39h	COMPARE	x [2]								8.2 (see page 160)
3Ah	COPY AND VERIFY	x [2]								8.4 (see page 160)
3Bh	WRITE BUFFER	x			x					8.64 (see page 381)
3Ch	READ BUFFER	x			x					8.25 (see page 309)
3Eh	READ LONG		x		x					8.31 (see page 322)
3Fh	WRITE LONG		x		x					8.65 (see page 386)
40h	CHANGE DEFINITION	x [2]								8.1 (see page 160)
41h	WRITE SAME (10)		x							8.67 (see page 388)

Table 64. Commands sorted by operation code (continued)

Op code [1]	Command name	Command type		Length						Reference
		All devices	Direct access devices	6 byte	10 byte	12 byte	16 byte	32 byte	Variable	
4Ch	LOG SELECT	x			x					8.9 (see page 200)
4Dh	LOG SENSE	x			x					8.10 (see page 203)
50h	XDWRITE (10)		x							8.71 (see page 393)
51h	XPWRITE (10)		x							8.75 (see page 399)
52h	XDREAD (10)		x							8.69 (see page 390)
53h	XDWRITEREAD (10)		x		x					8.73 (see page 396)
55h	MODE SELECT (10)		x		x					8.12 (see page 232)
56h	RESERVE (10)		x [2]		x					8.40 (see page 348)
57h	RELEASE (10)		x [2]		x					8.35 (see page 330)
5Ah	MODE SENSE (10)		x		x					8.14 (see page 281)
5Eh	PERSISTENT RESERVE IN		x							8.16 (see page 285)
5Fh	PERSISTENT RESERVE OUT		x							8.17 (see page 292)
7Fh	XDREAD (32)		x					x	0003h	8.70 (see page 391)
7Fh	XDWRITE (32)		x					x	0004h	8.72 (see page 394)
7Fh	XPWRITE (32)		x					x	0006h	8.76 (see page 400)
7Fh	XDWRITEREAD (32)		x					x	0007h	8.74 (see page 397)
88h	READ (16)		x							8.24 (see page 307)
8Ah	WRITE (16)		x				x			8.60 (see page 373)
8Eh	WRITE AND VERIFY (16)		x				x			8.63 (see page 379)
8Fh	VERIFY (16)		x				x			8.56 (see page 365)
90h	PREFETCH (16)		x				x			8.19 (see page 299)
91h	SYNCHRONIZE CACHE (16)		x							8.52 (see page 358)
92h	LOCK-UNLOCK CACHE (16)		x				x			8.8 (see page 198)
93h	WRITE SAME (16)		x				x			8.67 (see page 388)
9Eh	READ CAPACITY (16)		x				x		10h	8.27 (see page 315)
A0h	REPORT LUNS		x							8.37 (see page 333)
A3h	REPORT DEVICE IDENTIFIER	x				x				8.36 (see page 331)
A4h	SET DEVICE IDENTIFIER	x				x				8.48 (see page 354)
A7h	MOVE MEDIUM									8.15 (see page 284)
A8h	READ (12)		x			x				8.23 (see page 305)
AAh	WRITE (12)		x			x				8.59 (see page 371)
AEh	WRITE AND VERIFY (12)		x			x				8.62 (see page 377)
AFh	VERIFY (12)		x			x				8.55 (see page 363)
B4h	READ ELEMENT STATUS		x							8.30 (see page 321)
B7h	READ DEFECT DATA (12)		x			x				8.29 (see page 319)

[1] Command operation code.

[2] Obsolete command.

[3] Formerly called READ EXTENDED.

8.1 CHANGE DEFINITION command (40h)

Obsolete. The CHANGE DEFINITION command was implemented only by SCSI-1/SCSI-2 devices.

8.2 COMPARE command (39h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.3 COPY command (18h)

Obsolete. If received, the drive sends a CHECK CONDITION status and a Sense Key of ILLEGAL REQUEST.

8.4 COPY AND VERIFY command (3Ah)

Obsolete. If received, a CHECK CONDITION status is sent.

8.5 FORMAT UNIT command (04h)

The FORMAT UNIT command (Table 65) ensures that the medium is formatted so all of the user addressable data blocks can be accessed. There is no guarantee that the medium has or has not been altered. In addition, the medium may be certified and control structures may be created for the management of the medium and defects.

For systems which support disconnection, the drive disconnects while executing the FORMAT UNIT command.

This command is implemented in the drive for mandatory features and a subset of the available optional features of the ANSI SCSI-2 specification as defined in the following paragraphs.

The drive allows a SCSI initiator port to specify (or not specify) sectors which are to be reallocated during the format process. The format parameters to be specified in the FORMAT UNIT command are defined in Table 66.

The FORMAT UNIT command shall be rejected with RESERVATION CONFLICT status if the logical unit is reserved or any extent reservation from any initiator is active in the specified logical unit.

IMPLEMENTORS NOTE: The MODE SELECT parameters (if any) should be set prior to issuing the FORMAT UNIT command.

Table 65. FORMAT UNIT command (04h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	0	0
1	RESERVED [1]		LONGLIST [2]	FMTDATA [3]	CMPLST [4]	DEFECT LIST FORMAT [5]		
2	0	0	0	0	0	0	0	0
3	INTERLEAVE [6]							
4								
5	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A LONGLIST bit of zero indicates that the defect list header follows the short format in Table 69. A LONGLIST bit of one indicates that the defect list header follows the long format in Table 69.

[3] A Format Data (FMTDATA) bit of one indicates that the FORMAT UNIT Parameter list (Table 67) is supplied during the DATA OUT phase. The DATA OUT phase consists of a defect list header (Table 69), followed by an initialization pattern descriptor (Table 70) (if any), followed by the defect descriptors. The format of the defect descriptor list is determined by the DEFECT LIST FORMAT field. A FMTDATA bit of zero indicates the DATA OUT phase shall not occur (no defect data shall be supplied by the initiator).

If the FMTDATA bit is zero the CMPLST bit is ignored. See Table 66 for further information.

[4] If FMTDATA bit is one, a Complete List (CMPLST) bit of one indicates the data supplied is to be the complete list of Growth defects. Any previous Growth defect data or Certification defect data shall be erased. The drive may add to this list as it formats the medium. The result is to purge any previous Growth or Certification defect list and to build a new defect list. A CMPLST bit of zero indicates the data supplied is in addition to the existing Growth defect list.

The use of the P list and C list defect is controlled by byte 1 of the defect list header (see Table 69).

[5] The DEFECT LIST FORMAT field specifies additional information related to the defect list (see Table 66 for further information).

[6] The INTERLEAVE field requests that logical blocks be related in a specific fashion to the physical blocks to facilitate data transfer speed matching. An interleave value of zero requests that the target use its default interleave. An INTERLEAVE value of one requests that consecutive logical blocks be placed in consecutive physical order. Values of two or greater indicate that one or more (respectively) physical blocks separate consecutive logical blocks. The drive implements an optional prefetch (“read look ahead”) function which reads a user-specified number of sectors into its buffer beyond and contiguous to the sectors requested by the READ command. This data is subsequently available for the next sequential READ command without re-access of the disc media thereby increasing performance and negating the need for an interleave during format if this prefetch option is enabled (see READ command, Section 8.21). Interleave values other than zero or one are vendor-specific.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

8.5.1 FORMAT UNIT parameter definition

The following definitions of flaw categories are supplied to help in understanding the alternatives listed in Table 66.

P Primary Defect (P type) flawed sectors are identified at the time of shipment in a list of defects (permanent flaws) supplied by Seagate and stored on the disc in an area that is not directly accessible by the user. (This list may be referred to as an ETF List.) This defect list is not modified or changed by the drive (or initiator) after shipment.

C Certification Defect (C type) flawed sectors are sectors that fail a Format Verify during the format function.

D Data Defect (D type) sectors are sectors identified in a list supplied to the target by the initiator during a DATA OUT phase of the current FORMAT UNIT command. The D List follows a four byte defect list header and is referred to as Defect Descriptor Bytes.

G Growth Defect (G type) flawed sectors contain medium flaws and have been reallocated as a result of receiving a REASSIGN BLOCKS command, or certification defects (C type) reallocated during a previous FORMAT UNIT command, or Data Defects (D type) reallocated during a previous FORMAT UNIT command or defects that have been automatically reallocated by the drive. This (G) list is recorded on the drive media and may be referenced for the current (and subsequent) FORMAT UNIT commands. This (G) list does not include the Primary (P) list of defects.

Table 66. FORMAT UNIT parameter definition (format variations)

CDB-Byte (see Table 65)					
FMT DATA	CMP LIST	Defect List Format			
Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Comments
0	X	X	X	X	Default format: No DATA OUT phase occurs. Drive reallocates all sectors in the P list plus any sector which fails the Format Verify phase (C type flaws). Any previous G list is erased.
Block format [2]					
1	0	0	X	X	Format with G and no D: A four byte Defect List Header must be sent by the initiator. No Defect Descriptors (D list) are sent by the initiator. The drive reallocates all sectors in the drive's current G list [1].
1	1	0	X	X	Format without G or D: A four byte Defect List Header must be sent by the initiator. No D list may be sent by the initiator. The drive erases any previous G list [1].
Bytes from Index format					
1	1	1	0	0	Format with D and without G. The initiator must send a four byte Defect List Header followed by a D list of the defects for the drive to reallocate. The D list must be in the bytes from Index format (see Table 74). The drive erases any previous G list [1].
1	0	1	0	0	Format with D and with G: The initiator must send a four byte Defect List Header followed by a D list of the defects for the drive to reallocate. The D list must be in the bytes from Index format (see Table 74). The drive also reallocates all sectors in the drive's current G list [1].
Physical Sector format					
1	1	1	0	1	Format with D and without G: The initiator must send a four byte Defect List Header followed by a D List of defects for the drive to reallocate. The D list must be in the Physical Sector format (see Table 75). The drive erases any previous G list [1].
1	0	1	0	1	Format with D and with G: The initiator must send a four byte Defect List Header followed by a D List of defects for drive to reallocate. The D list must be in the Physical Sector format (see Table 75). The drive also reallocates all sectors in the drive's current G list [1].

[1] Byte one of the Defect List Header determines whether the P and C defects are reallocated (see Table 69 and 69).

[2] See individual drive's Product Manual, Volume 1, for support/nonsupport.

8.5.2 FORMAT UNIT parameter list

The defect list shown in Table 69 contains a four byte header followed by one or more defect descriptors (Table 73). The Defect List Length in each table specifies the total length in bytes of the defect descriptors that follow. In Table 69 the Defect List Length is equal to eight times the number of defect descriptors.

Table 67. FORMAT UNIT parameter list

Bit Byte	7	6	5	4	3	2	1	0
[1]	DEFECT LIST HEADER							
[2]	INITIALIZATION PATTERN DESCRIPTOR							
[3]	DEFECT DESCRIPTOR(S)							

[1] See Tables 68 and 69.

[2] See Table 70.

[3] See Tables 73, 74, and 75.

8.5.2.1 Defect List header

Table 68. Short Defect List header

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	FOV [1]	DPRY [2]	DCRT [3]	STPF [4]	IP [5]	DSP [6]	IMMED [7]	VS [8]
2	(MSB)	DEFECT LIST LENGTH [9]						
3								(LSB)

Table 69. Long Defect List header

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	FOV [1]	DPRY [2]	DCRT [3]	STPF [4]	IP [5]	DSP [6]	IMMED [7]	VS [8]
2	Reserved							
3	Reserved							
4	(MSB)							
5	DEFECT LIST LENGTH [9]							
6								
7								(LSB)

[1] Format Options Valid (FOV). A format options valid (FOV) bit of zero indicates that the device server shall use its default settings for the DPRY, DCRT, STPF, IP and DSP bits. If FOV is zero, the application client shall set these bits to zero. If FOV is zero and any of these 5 bitd are not zero, the device server shall terminate the command with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST with the additional sense code set to INVALID FIELD IN PARAMETER LIST. A FOV bit of one indicates that the device server shall examine the setting of the DPRY, DCRT, STPF, IP and DSP bits.

[2] Disable Primary (DPRY). If one, flaws in the drive P list are not reallocated during formatting. This means existing reallocations of the P list are canceled and no new reallocations made during formatting. The P list is retained. Some Seagate drives do not support a DPRY bit of one. See individual drive’s Product Manual, Volume 1.

If zero, flaws in the drive P list are reallocated during formatting. A Check Condition is sent in the status if the P list cannot be found by the drive.

[3] Disable Certification (DCRT). If one, the drive does not perform a verify function during formatting (thus no C list for this format is created or reallocated).

If zero, the drive performs a verify function during formatting and reallocates any sector that fails the verify (i.e., a C list is created and these flaws reallocated).

On drives that have MR heads, the DCRT bit is always interpreted as 0 when the user sets the IP bit (see Table 69) and specifies a format pattern of greater than 1 byte (see Table 70), and/or if the IP Modifier bits are set to 01 or 10 (see Table 70). This is done since the user format pattern and/or IP header is written onto the media during the format certification phase. Because of this the user may see G list entries added to the defect list even though the DCRT bit was set to 1.

[4] Stop Format (STPF). If one, formatting is terminated with a CHECK CONDITION status if an error is encountered while accessing either the P or G defect list. The sense key is set to Medium Error and the additional sense code shall be set to either DEFECT LIST NOT FOUND or DEFECT LIST ERROR.

If zero, formatting is not terminated if an error is encountered while accessing either the P or G defect list.

IMPLEMENTORS NOTE: The use of the FMTDATA bit, the CMPLST bit, and the defect header allow the application client to control the source of the defect lists used by the FORMAT UNIT command. Setting the DEFECT LIST LENGTH to zero allows the application client to control the use of PLIST and CLIST without having to specify a DLIST.

[5] Initialization Pattern (IP). An IP bit of one indicates that an initialization pattern descriptor is included in the FORMAT UNIT parameter list immediately following the defect list header. An IP bit of zero indicates that an initialization pattern descriptor is not included and that the target uses its default initialization pattern.

The initialization pattern type field indicates the type of pattern the target shall use to initialize each logical block within the initiator accessible portion of the medium (Table 72). All bytes within a logical block shall be written with the initialization pattern. The initialization pattern is modified by the IP modifier field as described in Table 71.

[6] Disable Saving Parameters (DSP). A DSP bit of one, specifies that the target shall not save the MODE SELECT savable parameters to non-volatile memory during the format operation. A DSP bit of zero specifies that the target shall save all the MODE SELECT savable parameters for all initiators to non-volatile memory during the format operation.

[7] Immediate (IMMED). An IMMED bit of zero indicates that status shall be returned after the first operation has completed. An Immed bit value of one indicates that the target shall return status as soon as the command descriptor block has been validated, and the entire defect list has been transferred. Check individual drive's Product Manual, Volume 1, for applicability of this feature.

[8] vendor-specific (VS). Not used.

[9] DEFECT LIST LENGTH. The length of any following D list (Defect Descriptors) must be equal to 8 times the number of sectors to be reallocated per Table 74 or Table 75.

Note. The initiator may not use any previously defined “C,” “G,” or “D” lists if the sector size (block length) has been changed.

8.5.2.2 Initialization Pattern Descriptor

The Initialization Pattern option specifies that the logical blocks contain the specified initialization pattern. The Initialization Pattern Descriptor (Table 70) is sent to the target as part of the FORMAT UNIT parameter list.

Table 70. Initialization Pattern Descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	IP MODIFIER [1]		Reserved					
1	PATTERN TYPE [2]							
2	(MSB)	INITIALIZATION PATTERN LENGTH [3]						
3								(LSB)
0-n	INITIALIZATION PATTERN [1]							

[1] INITIALIZATION PATTERN (IP) and IP MODIFIER. The Initialization Pattern is modified in accordance with the specification of the IP MODIFIER field given in Table 71.

Table 71. Initialization Pattern and IP Modifier

IP Modifier		Description
0	0	No header. The target shall not modify the initialization pattern.
0	1	The target shall overwrite the initialization pattern to write the LOGICAL BLOCK ADDRESS in the first four bytes of the logical block. The LOGICAL BLOCK ADDRESS shall be written with the most significant byte first.
1	0	The target shall overwrite the initialization pattern to write the LOGICAL BLOCK ADDRESS in the first four bytes of each physical block contained within the logical block. The lowest numbered logical block or part thereof that occurs within the physical block is used. The LOGICAL BLOCK ADDRESS shall be written with the most significant byte first.
1	1	Reserved.

[2] Pattern Type. Table 72 gives pattern type information.

Table 72. Initialization Pattern Type

Pattern Type	Description
00h	Use default pattern. If the initialization pattern length is not zero, this is an error and the drive shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense error code shall be set to INVALID FIELD IN PARAMETER LIST.
01h	Use pattern supplied by host computer. If the initialization pattern length is zero, this is an error and the drive shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense error code shall be set to INVALID FIELD IN PARAMETER LIST.
02h–7Fh	Reserved.
80h–FFh	Vendor-specific.

[3] The Initialization Pattern Length field indicates the number of bytes contained in the initialization pattern. If the length exceeds the current logical block size this is an error and the drive shall terminate the com-

mand with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense error code shall be set to Invalid Field In Parameter List. The initialization pattern is sent by the host for the drive to write in each logical block by the FORMAT command.

8.5.2.3 Defect List formats

This section describes the format of the defect list that follows the Defect List Header described in Table 69. Three formats are possible:

- Block format (Seagate drives do not support the block format, except as a customer special)
- Bytes from Index format (see Table 74)
- Physical Sector format (see Table 75)

IMPLEMENTORS NOTE: The selected reporting format accounts for variables that impact the information in the returned data. For example, the specific location of a defect, while constant in angular and radial location on the block device, may change in reported location if a format operation with different geometry parameters is performed. It is the responsibility of the initiator to use a defect list format appropriate for the intended operation with the current or future geometry parameters. If the target is able to detect that the selected defect list format would provide inconsistent results, the target may return CHECK CONDITION status.

Table 73. Defect Descriptors

0-n	Defect Descriptor Bytes (Refer to Tables 74 and 75 for formats)

Table 74. Defect Descriptor Bytes—Bytes from Index format

Byte Number	Description
0	CYLINDER NUMBER OF DEFECT (MSB)
1	CYLINDER NUMBER OF DEFECT
2	CYLINDER NUMBER OF DEFECT (LSB)
3	HEAD NUMBER OF DEFECT
4	DEFECT BYTES FROM INDEX (MSB)
5	DEFECT BYTES FROM INDEX
6	DEFECT BYTES FROM INDEX
7	DEFECT BYTES FROM INDEX (LSB)

For defects to be specified in the Bytes from Index format, the Defect List Format field (Byte 1, bits 2, 1, 0) must be 100 (binary) (see Tables 65 and 66).

Each Defect Descriptor for the Bytes from Index format specifies the beginning of an eight bit (1 byte) defect location on the medium. Each defect descriptor is comprised of the cylinder number of the defect, the head number of the defect, and the number of bytes from Index to the defect location (defect bytes from Index).

The Defect Descriptors shall be in ascending order. The drive may return CHECK CONDITION if the defect descriptors are not in ascending order. For determining ascending order, the CYLINDER NUMBER OF DEFECT is considered the most significant part of the address and the DEFECT BYTES FROM INDEX is considered the least significant part of the address.

If the DEFECT BYTES FROM INDEX has a value of FFFFFFFFh, this indicates that the entire track shall be considered defective.

Table 75. Defect Descriptor Bytes—Physical Sector format

Byte Number	Description
0	CYLINDER NUMBER OF DEFECT (MSB)
1	CYLINDER NUMBER OF DEFECT
2	CYLINDER NUMBER OF DEFECT (LSB)
3	HEAD NUMBER OF DEFECT
4	DEFECT SECTOR NUMBER (MSB)
5	DEFECT SECTOR NUMBER
6	DEFECT SECTOR NUMBER
7	DEFECT SECTOR NUMBER (LSB)

Information in this table is repeated for each defect.

For defects to be specified in the Physical Sector format, the Defect List format field (Byte 1, bits 2, 1, 0) must be 101 (binary). See Tables 65 and 66.

Each Defect Descriptor for the Physical Sector format specifies a sector size defect location comprised of the cylinder number of the defect, the head number of the defect, and the defect sector number.

The Defect Descriptors shall be in ascending order. The drive may return CHECK CONDITION if the defect descriptors are not in ascending order. For determining ascending order, the CYLINDER NUMBER OF DEFECT is considered the most significant part of the address and the DEFECT SECTOR NUMBER is considered the least significant part of the address.

A defect’s sector number of FFFFFFFFh indicates that the entire track shall be considered defective.

8.6 INQUIRY command (12h)

The INQUIRY command requests that information regarding parameters of the target and a component logical unit be sent to the initiator. Options allow the initiator to request additional information about the target and logical unit or information about SCSI commands supported by the target (see Section 8.6.3). Several INQUIRY commands may be sent to request the vital product data pages instead of the standard data shown in Table 77.

The INQUIRY command shall return CHECK CONDITION status only when the target is unable to return the requested INQUIRY data.

Table 76. INQUIRY command (12h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	1	0	0	1	0
1	Reserved						CMDDT [2]	EVPD [1]
2	PAGE OR OPERATION CODE [3]							
3	Reserved							
4	ALLOCATION LENGTH IN BYTES [4]							
5	CONTROL [5]							

[1] An Enable Vital Product Data (EVPD) bit of one specifies that the target shall return the optional vital product data specified by the PAGE OR OPERATION CODE field. If the logical unit does not support vital product data and this bit is set to one, the target shall return CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB.

[2] A Command Support Data (CMDDT) bit of one specifies that the target shall return the optional command support data specified by the Page or OPERATION CODE field. If the target does not support returning command data and this bit is set to one, the target shall return CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB. Details of the command support data may be found in Section 8.6.4.

Note. A SCSI initiator may receive a CHECK CONDITION status response with the sense key set to ILLEGAL REQUEST upon sending an INQUIRY command with the CMDDT bit set to 1 to some SCSI-2 target, since this bit was reserved in SCSI-2.

If both the EVPD and CMDDT bits are zero, the target shall return the standard INQUIRY data (see Section 8.6.1). If the Page or OPERATION CODE field is not zero when both EVPD and CMDDT are zero, the target shall return CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB.

If both the EVPD and CMDDT bits are one, the target shall return CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB. When the EVPD bit is one, the Page or OPERATION CODE field specifies which page of vital product data information the target shall return (see Section 8.6.3).

When the CMDDT bit is one, the PAGE OR OPERATION CODE field specifies the SCSI operation code for which target shall return command support data (see Section 8.6.4).

[3] The PAGE OR OPERATION CODE field specifies the page of vital product data information that the drive shall return. If the EVPD bit is zero and the page code field is not zero the drive shall return CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB.

[4] The ALLOCATION LENGTH specifies the number of bytes that the initiator has allocated for returned data. The drive shall terminate the DATA IN phase when ALLOCATION LENGTH bytes have been transferred or when all available data have been transferred to the initiator, whichever is less.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

If an INQUIRY command is received from a SCSI initiator port with a pending unit attention condition (i.e., before the target reports CHECK CONDITION status), the target shall perform the INQUIRY command and shall not clear the UNIT ATTENTION condition (see SAM-2, T10/1157D).

The INQUIRY data should be returned even though the target is not ready for other commands. To minimize delays after a HARD RESET or POWER-UP condition, the standard INQUIRY data should be available without incurring any media access delays. If the target does store some of the INQUIRY data on the media, it may return zeros or ASCII spaces (20h) in those fields until the data is available from the media.

The INQUIRY data may change as the target executes its initialization sequence. For example, the target may contain a minimum command set in its non-volatile memory and may load its final firmware from the media when it becomes ready. After the target has loaded the firmware, it may support more options and therefore return different supported options information in the INQUIRY data.

If the standard INQUIRY data changes for any reason, the target shall generate a UNIT ATTENTION condition for all initiators (see SAM-2). The target shall set the additional sense code to INQUIRY DATA HAS CHANGED.

Note. The INQUIRY command may be used by an initiator after a HARD RESET or POWER-UP condition to determine the device types for system configuration.

8.6.1 Drive standard INQUIRY data

The drive standard INQUIRY data contains 36 required bytes, followed by a number of bytes of drive specific data that is drive dependent (see individual drive’s Product Manual, Volume 1). The standard INQUIRY data is given in Table 77.

Table 77. Standard INQUIRY Data format

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	RMB [2]	Reserved						
2	VERSION [3]							
3	0 Obsolete	0 Obsolete	0 NORM ACA [4]	0 HISUPPORT [5]	RESPONSE DATA FORMAT [6]			
4	ADDITIONAL LENGTH (n–4) [7]							
5	SCCS [8]	ACC [9]	ALUA [10]	3PC [11]	Reserved			
6	BQUE [12]	ENCSERV [13]	Vendor Specific [14]	MULTIP [15]	MCHANGR [16]	Obsolete [17]	Obsolete [18]	ADDR16 [19]
7	RELADR [20]	Obsolete [21]	WBUS16 [22]	SYNC [23]	LINKED [24]	Obsolete [25]	CMDQUE [26]	Vendor specific [27]
8 : 15	VENDOR IDENTIFICATION [28]							
16 : 31	PRODUCT IDENTIFICATION [29]							
32 : 35	PRODUCT REVISION LEVEL [30]							
36 : 43	Vvendor-specific (formerly Product Description field in SCSI-2) [31]							
44 : 55	Vendor-specific (00h) [32]							
56	Reserved (00h) [33]				CLOCKING [34]		QAS [35]	IUS [36]
57	Reserved							
58	(MSB)	VERSION DESCRIPTOR 1 [38]						
59								(LSB)

Bit Byte	7	6	5	4	3	2	1	0
60 : 71	:							
72	(MSB)	VERSION DESCRIPTOR 8 [38]						
73								(LSB)
74 : 95	Reserved							
Vendor-Specific Parameters								
96 : n	Copyright Notice [37]							

[1] The PERIPHERAL QUALIFIER and PERIPHERAL DEVICE TYPE fields identify the device currently connected to the logical unit. If the target is not capable of supporting a device on this logical unit, the target shall set this field to 7Fh (PERIPHERAL QUALIFIER set to 011b and PERIPHERAL DEVICE TYPE set to 1Fh). The PERIPHERAL QUALIFIER is defined in Table 78 and the peripheral device type is defined in Table 79.

Table 78: Peripheral qualifier

Qualifier	Description
000b	The specified peripheral device type is currently connected to this logical unit. If the target is unable to determine whether or not a physical device is currently connected, it also shall use this peripheral qualifier when returning the INQUIRY data. This peripheral qualifier does not mean that the device is ready for access by the initiator.
001b	The target is capable of supporting the specified peripheral device type on this logical unit. However, the physical device is not currently connected to this logical unit.
010b	Reserved.
011b	The target is not capable of supporting a physical device on this logical unit. For this peripheral qualifier the peripheral device type shall be set to 1Fh to provide compatibility with previous versions of SCSI. All other peripheral device type values are reserved for this peripheral qualifier.
1XXb	Vendor-specific.

Table 79: Peripheral device type

Code	Doc. [a]	Description
00h	SBC	Direct access device (e.g., magnetic disc)
01h	SSC	Sequential access device
02h	SSC	Printer device
03h	SPC-3	Processor device
04h	SBC	Write-once device
05h	MMC-2	CD-ROM device
06h	SCSI-2	Scanner device
07h	SBC	Optical memory device
08h	SMC	Medium changer device
09h		Communications device
0Ah- 0Bh		Defined by ASC ITS (Graphic arts pre-press device)
0Ch	SCC-2	Storage array controller device
0Dh	SES	Enclosure services device
0Eh	RBC	Simplified direct access device
0Fh	OCRW	Optical card reader/writer device
10h		Reserved [b]
11h	OSD	Object-based Storage Device
12h	ADC	Automation/Drive Interface
13h- 1Dh		Reserved
1Eh		Well known logical unit [c]
1Fh		Unknown or no device type

[a] 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 listed standards.

[b] Type code 10h is reserved for use by Bridging Expanders.

[c] All well known logical units use the same peripheral device type code.

[2] RMB. A removable medium bit of zero indicates that the medium is not removable. A RMB bit of one indicates that the medium is removable.

[3] The ANSI-APPROVED VERSION field indicates the implemented version of this manual and is defined below in Table 80.

Table 80: ANSI-approved version

Code	Description
00h	The device does not claim conformance to any standard
02h	Obsolete
03h	The device complies to ANSI X3.301:1997 (SPC)
04h	The device complies to ANSI X3.351:200x (SPC-2)
05h	The device complies to T10/1416-D

Code	Description	Code	Description
1h	Obsolete	6h-7h	Reserved
08h-0Ch	Obsolete	0Dh-3Fh	Reserved
40h-44h	Obsolete	45h-47h	Reserved
48h-4Ch	Obsolete	4Dh-7Fh	Reserved
80h-84h	Obsolete	85h-87h	Reserved
88h-8Ch	Obsolete	8Dh-FFh	Reserved

[4] The Normal ACA Supported (NORMACA) bit of one indicates that the target supports setting the NACA bit to one in the Control Byte of the CDB (as defined in ANSI document SCSI Architecture Model-2, SAM-2, T10/1157). A NormACA bit of zero indicates that the target does not support setting the NACA bit to one.”

[5] A Hierarchical Support (HISUPPORT) bit of zero indicates the target does not use the hierarchical addressing model to assign LUNs to logical units. A HiSupport bit of one indicates the target uses the hierarchical addressing model to assign LUNs to logical units. When the HiSupport bit is one, the target shall support the REPORT LUNS command (see Section 8.37).

[6] A RESPONSE DATA FORMAT value of two indicates the INQUIRY data format is as specified in this manual. A RESPONSE DATA FORMAT value of one indicates compatibility with some products that were designed prior to the development of the ANSI SCSI Controller Command Set-2 standard. RESPONSE DATA FORMAT values greater than two are reserved.

[7] The ADDITIONAL LENGTH field shall specify the length in bytes of the parameters. If the ALLOCATION LENGTH of the command descriptor block is too small to transfer all of the parameters, the Additional Length is not adjusted to reflect the truncation.

[8] An SCC Supported (SCCS) bit of one indicates that the device contains an embedded storage array controller component. See SCC-2 (SCSI Controller Command Set-2, T10/1225-D) for details about storage array controller devices. An SCCS bit of zero indicates that the device does not contain an embedded storage array controller component.

[9] An Access Controls Coordinator (ACC) bit of one indicates that the device contains an access controls coordinator that may be addressed through this logical unit. An ACC bit of zero indicates that no access controls coordinator may be addressed through this logical unit. If the device contains an access controls

coordinator that may be addressed through any logical unit other than the ACCESS CONTROLS well known logical unit, the ACC bit shall be set to one for LUN 0.

[10] The contents of the Asymmetric Logical Unit Access (ALUA) field (see table 81) indicate the support for asymmetric logical unit access.

Table 81. ALUA field contents

Value	Description
00b	The SCSI target device does not support asymmetric logical unit access or supports a form of asymmetric access that is vendor specific. Neither the REPORT TARGET GROUPS nor the SET TARGET GROUPS commands is supported.
01b	Only implicit asymmetric logical unit access (see 5.7.6) is supported. The SCSI target device is capable of changing target port asymmetric access states without a SET TARGET PORT GROUPS command. The REPORT TARGET PORT GROUPS command is supported and the SET TARGET PORT GROUPS command is not supported.
10b	Only explicit asymmetric logical unit access (see 5.7.7) is supported. The SCSI target device only changes target port asymmetric access states as requested with the SET TARGET PORT GROUPS command. Both the REPORT TARGET PORT GROUPS command and the SET TARGET PORT GROUPS command are supported.
11b	Both explicit and implicit asymmetric logical unit access are supported. Both the REPORT TARGET PORT GROUPS command and the SET TARGET PORT GROUPS commands are supported.

[11] A Third-Party Copy (3PC) bit of one indicates that the device supports third-party copy commands such as the EXTENDED COPY command. A 3PC bit of zero indicates that the device does not support such commands.

[12] The Basic Queuing (BQUE) bit shall be zero if the CmdQue (see note [26]) bit is one. When the CmdQue bit is zero, the BQue bit shall have the following meaning. A BQue bit of zero indicates that the device does not support tagged tasks (command queuing) for this logical unit. A value of one indicates that the device supports, for this logical unit, the basic task management model defined by ANSI document SCSI Architecture Model-2 (SAM-2), T10/1157.

[13] An Enclosure Services (ENCSERV) bit of one indicates that the device contains an embedded enclosure services component. See ANSI Document SCSI Enclosure Services Command Set, NCITS 305-199X, T10/1212-D for details about enclosure services, including a device model for an embedded enclosure services device. An ENCSERV bit of zero indicates that the device does not contain an embedded enclosure services component.

[14] vendor-specific. Dual port A-B on some drives.

[15] A Multi Port (MULTIP) bit of one shall indicate that this is a multi-port (2 or more ports) device and conforms to the SCSI multi-port device requirements found in the applicable standards. A value of zero indicates that this device has a single port and does not implement the multi-port requirements.

[16] A Medium Changer (MCHANGR) bit of one indicates that the device is embedded within or attached to a medium transport element. See SCSI Medium Changer Commands (SMC), T10/999D, for details about medium changers, including a device model for an attached medium changer device. The MCHNGR bit is valid only when the RMB bit is equal to one. A MCHNGR bit of zero indicates that the device is not embedded within or attached to a medium transport element.

[17] Obsolete. Formerly AckReqQ.

[18] Obsolete. Formerly ADR32.

[19] Table 82 defines the relationships between the Wide SCSI Address 16 (ADDR16) and Wide Bus 16 (WBUS16) bits.

Table 82: Maximum logical device configuration table

ADDR 16	WBUS16	Description
0	0	8-bit wide data path on a single cable with 8 SCSI IDs supported
0	1	16-bit wide data path on a single cable with 8 SCSI IDs supported
1	1	16-bit wide data path on a single cable with 16 SCSI IDs supported

[20] A Relative Address (RELADR) bit of one indicates that the drive supports the relative addressing mode. If this bit is set to one, the Linked command (Linked) bit shall also be set to one since relative addressing can only be used with linked commands. A RELADR bit of zero indicates the drive does not support relative addressing.

[21] Obsolete. Formerly Wide Bus 32 (WBus32).

[22] A Wide Bus 16 (WBUS16) bit of one indicates that the target supports 16-bit wide data transfers. A value of zero indicates that the device does not support 16-bit wide data transfers (see Table 82).

[23] A Synchronous Transfer (SYNC) bit of one indicates that the drive supports synchronous data transfer. A value of zero indicates the drive does not support synchronous data transfer.

[24] A Linked Command (LINKED) bit of one indicates that the drive supports linked commands (see SAM-2). A value of zero indicates the drive does not support linked commands.

[25] Obsolete. Formerly Transfer Disable (TRNDIS).

[26] A Command Queuing (CMDQUE) bit of one indicates that the device supports tagged tasks (command queuing) for this logical unit (see ANSI document SCSI Architecture Model-2, T10/1157). A value of zero indicates the target may support tagged tasks for this logical unit (see note [12]). Table 83 summarizes the relationship of the BQUE and CMDQUE bits.

Table 83. Relationship of BQUE and CMDQUE bits

BQu e	CmdQu e	Description
0	0	No command queuing of any kind supported.
0	1	Command queuing with all types of task tags supported.
1	0	Basic task set model supported (see SAM-2).
1	1	Illegal combination of BQue and CmdQue bits.

[27] Vendor-specific bit.

[28] The VENDOR IDENTIFICATION field contains eight bytes of ASCII data identifying the vendor of the product. The data shall be left aligned within this field.

Note. It is intended that this field provide a unique vendor identification of the manufacturer of the SCSI device. In the absence of a formal registration procedure, T10 maintains a list of vendor identification codes in use. Vendors are requested to voluntarily submit their identification codes to T10 to prevent duplication of codes (see Annex D). For Seagate drives, the Vendor Identification field contains the ASCII data giving vendor name (“SEAGATE”).

This bit was formerly used for SOFT RESET.

[29] The PRODUCT IDENTIFICATION field contains ASCII data giving the drive model number. The data shall be left-aligned within this field.

[30] The PRODUCT REVISION LEVEL field contains the four bytes of ASCII data “XXXX,” where the value of “XXXX” is determined by each device manufacturer.

[31] PRODUCT DESCRIPTION field. In some models (SACI-2), this field contained the Drive Serial Number field, containing the 8 bytes of ASCII data “XXXXXXXX,” where the value XXXXXXXX is the drive serial number. In newer models it’s use is vendor-specific.

[32] Vendor specific.The area from byte 44 through byte 55 are vendor specific.

[33] Reserved.

[34] CLOCKING. The CLOCKING field shall not apply to asynchronous transfers and is defined in Table 84.

Table 84: Clocking field

Code	Description
00b	Indicates the target supports only Single Transition (ST)—the latching of data only on the assertion edge of the REQ(ACK) signal.
01b	Indicates the target supports only Double Transition (DT)—the latching of data on both the assertion edge and the negated edge of the REQ(ACK) signal.
10b	Reserved.
11b	Indicates the target supports ST and DT.

[35] QAS. A Quick Arbitrate Supported (QAS) bit of one indicates that the target supports the quick arbitrate feature. A value of zero indicates that the target does not support the quick arbitrate feature.

[36] IUS. Information Unit Supported.

[37] Copyright Notice field. This field contains 48 bytes of ASCII data “Copyright (c) XXXX Seagate All rights reserved,” where “XXXX” indicated the current year.

[38] VERSION DESCRIPTOR 2 through 8 (bytes 60-71). The VERSION DESCRIPTOR fields provide for identifying up to eight standards to which the device claims conformance. The value in each PAGE OR OPERATION CODE field shall be selected from Table 80 of SCSI Primary Commands-3 (SPC-3), T10/1416D. All version descriptor values not listed in Table 80 are reserved. Technical Committee T10 of NCITS maintains an electronic copy of the information in Table 80 on its world wide web site (http://www.t10.org/). In the event that the T10 world wide web site is no longer active, access may be possible via the NCITS world wide web site (http://www.ncits.org) or the ANSI world wide web site (http://www.ansi.org). It is recommended that the first version descriptor be used for the SCSI architecture standard, followed by the physical standard, followed by the physical/mapping protocol if any, followed by the appropriate SPC version, followed by the device type command set, followed by a secondary command set if any. The four and a half page Table 80 will not be reproduced here. Consult the above mentioned sources for the list of Vendor Descriptor values.

8.6.2 SCSI Parallel Interface (SPI) specific INQUIRY data

Refer back to Table 77. Portions of bytes 6 and 7 and all of byte 56 of the standard INQUIRY data shall be used only by the SCSI Parallel Interface. These fields are noted in Table 77. This section describes details on how the SPI-specific fields relate to the SCSI Parallel Interface see SPI-n (where n is 2 or greater). Table 85 shows just the SPI-specific standard INQUIRY fields. The definitions of the SCSI Parallel Interface specific fields shall be as follows.

Table 85. SPI-specific standard INQUIRY bits

Bit Byte	7	6	5	4	3	2	1	0
6	see Table 77							ADDR16 [1]
7	see Table 77		WBUS16 [2]	SYNC [3]	see Table 77	Obsolete [4]	see Table 77	
: : :	: : :							
56	Reserved				CLOCKING [5]		QAS [6]	IUS [7]

[1] A Wide SCSI Address 16 (ADDR16) bit of one indicates that the target supports 16-bit wide SCSI addresses. A value of zero indicates that the device does not support 16-bit wide SCSI addresses.

[2] A Wide Bus 16 (WBUS16) bit of one indicates that the target supports 16-bit wide data transfers. A value of zero indicates that the device does not support 16-bit wide data transfers. Table 82 defines the relationships between the ADDR16 and WBUS16 bits.

[3] A Synchronous Transfer (SYNC) bit of one indicates that the target supports synchronous data transfer. A value of zero indicates the device does not support synchronous data transfer.

[4] Obsolete. Formerly Transfer Disable (TRANDIS).

[5] The CLOCKING field shall not apply to asynchronous transfers and is defined in Table 84.

[6] A Quick Arbitrate Supported (QAS) bit of one indicates that the target supports the quick arbitrate feature. A value of zero indicates that the target does not support the quick arbitrate feature.

[7] An Information Unit Supported (IUS) bit of one indicates that the target supports information units. A value of zero indicates that the target does not support information units.

Note. The acronyms ST and DT and the terms “quick arbitrate” and “information units” are defined in Section 6 of this manual and in SCSI Parallel Interface (SPI-4).

8.6.3 Vital Product Data pages

This section describes the vital product data page structure and the vital product data pages that are applicable to Seagate devices. These pages are optionally returned by the INQUIRY command and contain specific product information about the drive. The vital product data includes vendor identification, product identification, unit serial numbers, device operating definitions, manufacturing data, field replaceable unit information, and other vendor-specific information.

The initiator requests the vital product data information by setting the EVPD bit to one and specifying the page code of the desired vital product data. If the target device does not implement the requested page, it shall return CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to Invalid Field in CDB.

Note. The target should have the ability to execute the INQUIRY command even when an error occurs that prohibits normal command execution. In such a case, CHECK CONDITION status should be returned for commands other than INQUIRY or REQUEST SENSE. The sense data returned may contain the field replaceable unit code. The vital product data may be obtained for the failing device using the INQUIRY command.

Note. This manual defines a format that allows device-independent initiator software to display the vital product data returned by the INQUIRY command. The contents of the data may be vendor-specific, and may be unusable without detailed information about the device.

Note. This manual does not define the location or method of storing the vital product data. The retrieval of the data may require completion of initialization operations within the device, that may induce delays before the data is available to the initiator. Time-critical requirements are an implementation consideration and are not addressed in this manual.

8.6.4 Command support data

Implementation of command support data is optional. The initiator may request the command support data information by setting the CMDDT bit to one and specifying the SCSI operation code of the desired CDB. If the target implements the requested SCSI operation code, it shall return the data defined in Table 86. If the target does not implement the requested SCSI operation code it shall return the peripheral qualifier and type byte and 001b in the Support field.

Table 86. Command support data format

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	Reserved					SUPPORT [2]		
2	VERSION [3]							
3	Reserved							
4	Reserved							
5	CDB SIZE (m – 5) [4]							
6 : m	CDB USAGE DATA [5]							

[1] The PERIPHERAL QUALIFIER and PERIPHERAL DEVICE TYPE fields identify the device currently connected to the logical unit. If the target is not capable of supporting a device on this logical unit, the target shall set this field to 7Fh (PERIPHERAL QUALIFIER set to 011b and PERIPHERAL DEVICE TYPE set to 1Fh). The PERIPHERAL QUALIFIER is defined in Table 78 and the PERIPHERAL DEVICE TYPE is defined in Table 79.

[2] Table 87 defines the values and meanings of the SUPPORT field.

Table 87: Support values and meanings

Support	Description
000b	Data about the requested SCSI operation is not currently available.
001b	The target does not support the tested SCSI operation code. All data after byte 1 is undefined.
010b	Reserved.
011b	The target supports the tested SCSI operation code in conformance with an SCSI standard. The data format conforms to the definition in Table 86.
100b	vendor-specific.
101b	The target supports the tested SCSI operation code in a vendor-specific manner. The data format conforms to the definition in Table 86.
110b	vendor-specific.
111b	Reserved.

If the SUPPORT field contains 000b, all data after byte 1 is not valid. One possible reason for SUPPORT being 000b is the target's inability to retrieve information stored on the media. When this is the case, a subsequent request for command support data may be successful.

[3] VERSION. See Table 77.

[4] The CDB SIZE field shall contain the number of bytes in the CDB for the operation code being queried, and the size of the CDB USAGE DATA field in the return data.

Note. The CDB SIZE field is provided primarily for the convenience of the initiator. In most cases, the size is known from the operation code group.

[5] The CDB USAGE DATA field shall contain information about the CDB for the operation code being queried. The first byte of the CDB usage data shall contain the operation code for the operation being queried. All bytes except the first byte of the CDB usage data shall contain a usage map for bits in the CDB for the operation code being queried.

The bits in the usage bit map have a one-for-one correspondence to the CDB for the operation code being queried. If the target evaluates a bit as all or part of a field in the CDB for the operation code being queried, the usage map shall contain a one in the corresponding bit position. If the target ignores or treats as reserved a bit in the CDB for the operation code being queried, the usage map shall contain a zero in the corresponding it position. The usage map bits for a given CDB field all shall have the same value.

Thus, the CDB usage bit map for the INQUIRY command for a target that implements command support data but not vital product data is: 12h, 02h, FFh, 00h, FFh, 07h. This example assumes that SAM-2 defines uses for only the low-order three bits of the Control byte. Note that the first byte contains the operation code and the remaining bytes contain the usage map.

8.6.5 Vital product data parameters

This section describes the optional vital product data page structure and the vital product data pages (see Table 88) that are applicable to all SCSI devices. These pages are optionally returned by the INQUIRY command and contain vendor-specific product information about a SCSI target port or logical unit. The vital product data may include vendor identification, product identification, unit serial numbers, device operating definitions, manufacturing data, field replaceable unit information, and other vendor-specific information.

This manual defines the structure of the vital product data, but not the contents.

Table 88. Vital Product Data page codes

Page code	Description
00h	Supported Vital Product Data pages [1]
01h - 7Fh	ASCII Information page
80h	Unit Serial Number page
81h	Obsolete (Implemented Operating Definition page)
82h	ASCII Implemented Operating Definition page
83h	Device Identification page
84h	Software Interface Identification page
85h - AFh	Reserved
B0 - BFh	See specific device type
C0 - FFh	vendor-specific page

[1] If a target supports any vital product data pages, it also shall support vital product data page code 00h.

8.6.5.1 ASCII Implemented Operating Definition page

The ASCII Implemented Operating Definition page (see Table 89) contains operating definition description data for all operating definitions implemented by the target.

Table 89. ASCII Implemented Operating Definition page

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (82h) [2]							
2	Reserved							
3	PAGE LENGTH (n-3) [3]							
4	ASCII OPERATING DEFINITION DESCRIPTION LENGTH (m-4) [4]							
5	ASCII OPERATING DEFINITION DESCRIPTION DATA [5]							
m								
m+1	VENDOR-SPECIFIC DESCRIPTION DATA [6]							
n								

[1] The PERIPHERAL QUALIFIER field and the PERIPHERAL DEVICE TYPE field are as defined in Table 77, Note [1].

[2] PAGE CODE field.

[3] The PAGE LENGTH field specifies the length of the following page data. If the ALLOCATION LENGTH is less than the length of the data to be returned, the page length shall not be adjusted to reflect the truncation.

[4] The ASCII OPERATING DEFINITION DESCRIPTION LENGTH field specifies the length in bytes of the ASCII OPERATING DEFINITION DESCRIPTION DATA field that follows. If the ALLOCATION LENGTH is less than the length of data to be returned, the ASCII operating definition description length shall not be adjusted to reflect the truncation. A value of zero in this field indicates that no ASCII operating definition description data is available.

[5] The ASCII OPERATING DEFINITION DESCRIPTION DATA field contains the ASCII operating definition description data for the target. The data in this field shall be formatted in lines (or character strings). Each line shall contain only graphic codes (i.e., code values 20h through 7Eh) and shall be terminated with a Null (00h) character. The text is vendor-specific.

[6] Vendor-Specific Description Data.

8.6.5.2 ASCII Information page

The ASCII Information page (see Table 90) contains information for the field replaceable unit code returned in the REQUEST SENSE data (see Section 8.38).

Table 90. ASCII Information page

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (01h - 7Fh) [2]							
2	Reserved							
3	PAGE LENGTH (n-3) [3]							
4	ASCII LENGTH (m-4) [4]							
5	ASCII INFORMATION [5]							
m								
m+1	Vendor-Specific Information [6]							
n								

[1] The PERIPHERAL QUALIFIER field and the Peripheral Device field are as defined in Table 77, Note [1].

[2] The PAGE CODE field contains the same value as in the Page or OPERATION CODE field of the INQUIRY command descriptor block (see Table 76) and is associated with the Field Replaceable Unit Code field returned by the REQUEST SENSE command. (See 8.38)

Note. The FIELD REPLACEABLE UNIT CODE field in the sense data provides for 255 possible codes, while the page code field provides for only 127 possible codes. Thus it is not possible to return ASCII information pages for the upper code values.

[3] The PAGE LENGTH field specifies the length of the following page data. If the ALLOCATION LENGTH of the command descriptor block is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The ASCII LENGTH field specifies the length in bytes of the ASCII Information field that follows. If the ALLOCATION LENGTH is less than the length of the data to be returned, the ASCII length shall not be adjusted to reflect the truncation. A value of zero in this field indicates that no ASCII information is available for the specified page code.

[5] The ASCII INFORMATION field contains ASCII information concerning the field replaceable unit identified by the page code. The data in this field shall be formatted in one or more lines (or character strings). Each line shall contain only graphic codes (i.e., code values 20h through 7Eh) and shall be terminated with a Null (00h) character.

[6] Vendor-Specific Information.

8.6.5.3 Device Identification page

The Device Identification page (see Table 91) provides the means to retrieve zero or more identification descriptors applying to the logical unit. Logical units may have more than one identification descriptor (e.g., if several types or associations of identifier are supported).

Device identifiers, if any, shall be assigned to the peripheral device (e.g., a disk drive) and not to the currently mounted media, in the case of removable media devices. Media identification is outside the scope of this manual. Operating systems are expected to use the device identifiers during system configuration activities to determine whether alternate paths exist for the same peripheral device.

Note. In the case of virtual logical units (e.g., volume sets as defined by SCC), the IDENTIFIER field (see Table 93) should be an IEEE Registered Extended name. The IEEE Registered Extended name has a code of 3h in the IDENTIFIER TYPE field and an NAA value of 0110b as defined in FC-PH, FC-PH-3 or FC-FS.

A SCSI target device may have more than one SCSI target device name if the SCSI target device supports multiple SCSI transport protocols. If the returned Device Identification VPD page contains any SCSI target device names it shall contain all the SCSI target device names.

Table 91. Device Identification page

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (83h) [2]							
2	MSB	PAGE LENGTH (n-3) [3]						
3								LSB
	IDENTIFICATION DESCRIPTOR LIST [4]							
4	IDENTIFICATION DESCRIPTOR (First) [5]							
:	:							
:	:							
:	:							
n	IDENTIFICATION DESCRIPTOR (Last) [5]							

[1] The PERIPHERAL QUALIFIER field and the PERIPHERAL DEVICE field are as defined in Table 77.

[2] PAGE CODE.

[3] PAGE LENGTH.

[4] The IDENTIFICATION DESCRIPTOR LIST provides a list of Identification Descriptor information, the format of which is given in Table 92.

[5] Each IDENTIFICATION DESCRIPTOR (see Table 92) contains information identifying the logical unit, physical device, or access path used by the command and returned parameter data.

Table 92: Identification Descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	PROTOCOL IDENTIFIER [a]				CODE SET [b]			
1	PIV [c]	Reserved	Association [d]		Identifier Type [e]			
2	Reserved							
3	Identifier Length (n – 3) [f]							
4	Identifier [g]							
n								

[a] The PROTOCOL IDENTIFIER field may indicate the SCSI transport protocol to which identification descriptor applies. If the ASSOCIATION field contains a value other than 1h or 2h or the PIV bit is set to zero, then the PROTOCOL IDENTIFIER field should be ignored. If the ASSOCIATION field contains a value of 1h or 2h and the PIV bit is set to one, then the PROTOCOL IDENTIFIER field shall contain one of the values shown in table 238 (see 7.5.1) to indicate the SCSI transport protocol to which identification descriptor applies.

[b] The CODE SET field specifies the code set used for the identifier field, as described in Table 93. This field is intended to be an aid to software that displays the identifier field.

Table 93: Code Set

Value	Description
0h	Reserved
1h	The Identifier field shall contain binary values
2h	The Identifier field shall contain ASCII graphic codes (i.e., code values 20h through 7Eh)
3h - Fh	Reserved

[c] A protocol identifier valid (PIV) bit of zero indicates the PROTOCOL IDENTIFIER field should be ignored. If the ASSOCIATION field contains a value of 1h or 2h then a PIV bit set to one indicates the PROTOCOL IDENTIFIER field contains a valid protocol identifier selected from the values shown in table 94. If the ASSOCIATION field contains a value other than 1h or 2h then the PIV bit should be ignored.

Table 94. PROTOCOL IDENTIFIER values

Protocol Identifier	Description	Protocol Standard
0h	Fibre Channel	FCP-2
1h	Parallel SCSI	SPI-4
2h	SSA	SSA-S3P
3h	IEEE 1394	SBP-2
4h	Remote Direct Memory Access (RDMA)	SRP
5h	Internet SCSI	iSCSI
6h	SAS Serial SCSI Protocol	SAS
7h - Fh	Reserved	

[d] The Association field indicates the entity that the Identification descriptor describes. If a physical or logical device returns an Identification descriptor with the Association field set to 0h, it shall return the same descriptor when it is accessed through any other path.

The Association field specifies the entity with which the Identifier field is associated, as described in Table 95.

Table 95: Association

Value	Description
0h	The Identifier field is associated with the addressed physical or logical device
1h	The Identifier field is associated with the port that received the request
2h	The Identifier field is associated with the SCSI target device that contains the addressed logical unit.
‘3h	Reserved

[e] The Identifier Type field specifies the format and assignment authority for the identifier, as described in tables 96 and 97.

Table 96: Identifier Type

Value	Description
0h	No assignment authority was used and consequently there is no guarantee that the identifier is globally unique (i.e., the identifier is vendor specific).
1h	The first 8 bytes of the identifier field are a Vendor ID (see annex C). The organization associated with the Vendor ID is responsible for ensuring that the remainder of the identifier field is unique. One recommended method of constructing the remainder of the identifier field is to concatenate the product identification field from the standard INQUIRY data field and the product serial number field from the unit serial number page.
2h	The identifier field contains a Canonical form IEEE Extended Unique Identifier, 64-bit (EUI-64). In this case, the identifier length field shall be set to 8. Note that the IEEE guide-lines for EUI-64 specify a method for unambiguously encapsulating an IEEE 48-bit identifier within an EUI-64.
3h	The identifier field contains an FC-PH, FC-PH3 or FC-FS Name_Identifier. Any FC-PH, FC-PH3 or FC-FS identifier may be used, including one of the four based on a Canonical form IEEE company_id.
4h	If the Association field contains 1h, the Identifier value contains a four-byte binary number identifying the port relative to other ports in the device using the values shown Table 97. The CODE SET field shall be set to 1h and the IDENTIFIER LENGTH field shall be set to 4h. If the ASSOCIATION field does not contain 1h, use of this identifier type is reserved.
5h	If the Association value is 1h, the Identifier value contains a four-byte binary number identifying the port relative to other ports in the device using the values shown Table 97. The CODE SET field shall be set to 1h and the IDENTIFIER LENGTH field shall be set to 4h. If the Association field does not contain 1h, use of this identifier type is reserved.
6h	If the Association value is 0h, the Identifier value contains a four-byte binary number identifying the port relative to other ports in the device using the values shown Table 97. The CODE SET field shall be set to 1h and the IDENTIFIER LENGTH field shall be set to 4h. If the Association field does not contain 0h, use of this identifier type is reserved.
7h	The MD5 logical unit identifier shall not be used if a logical unit provides unique identification using identifier types 2h or 3h. A bridge device may return a MD5 logical unit identifier type for that logical unit that does not support the Device Identification VPD page.
8h - Fh	Reserved.

Table 97: Relative Port identifier values

Value	Description
0h	Reserved
1h	Relative port 1, also known as port A
2h	Relative port 2, also known as port B
3h - 7FFFFFFFh	Relative port 3 through 2 147 483 647
80000000h - FFFFFFFFh	

[f] The Identifier Length field specifies the length in bytes of the Identifier field. If the ALLOCATION LENGTH field of the command descriptor block is too small to transfer all of the identifier, the identifier length shall not be adjusted to reflect the truncation.

[g] The Identifier field contains the identifier as described by the Association, Identifier Type, CODE SET, and Identifier Length fields. The example described in this section and shown in Table 98 is not a normative part of this manual. This example of a complete device identification VPD page assumes that the product is a direct-access device with an T10 Vendor ID of “XYZ_Corp,” a product identification of “Super Turbo Disk,” and a product serial number of “2034589345.” Furthermore, it is assumed that the manufacturer has been assigned a 24-bit IEEE company_id of 01ABCDh by the IEEE Registration Authority Committee and that the manufacture has assigned a 24-bit extension_identifier of 234567h to this logical unit. The combined 48-bit identifier is reported in the 64-bit format as defined by the IEEE 64-bit Global Identifier (EUI-64) standard. The data returned in the device identification VPD page for this logical unit is shown in Table 98.

Table 98. Device Identification page example

Bytes	Hexadecimal values	ASCII values
00–15	00 83 00 32 02 01 00 22 58 59 5A 5F 43 6F 72 70	...2...XYZ_Corp [5]
16–31	53 75 70 65 72 20 54 75 72 62 6F 20 44 69 73 6B	Super Turbo Disk
32–47	32 30 33 34 35 38 39 33 34 35 01 02 00 08 01 AB	2034589345.......
48–53	CD FF FF 23 45 67	......

Notes.

1. Non-printing ASCII characters are shown as “.”.
2. Byte 00 is the beginning of the VPD page (see Table 91).
3. Byte 04 is the beginning of the Identification descriptor for the Vendor ID based identifier (Identifier type 1 see Table 96).
4. Byte 42 is the beginning of the Identification Descriptor for the EUI-64 identifier (Identifier type 2, see Table 96).
5. For Seagate devices, this will say “Seagate.”

8.6.5.4 Supported Vital Product Data pages

This section contains a list of the vital product data page codes supported by the target or logical unit (see Table 99). If a target supports any vital product data pages, it also shall support this vital product data page.

Table 99. Supported Vital Product Data pages

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (80h)[2]							
2	Reserved							
3	PAGE LENGTH (n-3) [3]							
4	SUPPORTED PAGE LIST [4]							
n								

[1] The PERIPHERAL QUALIFIER field and the PERIPHERAL DEVICE TYPE field are defined in Table 77, Note [1].

[2] PAGE CODE field.

[3] The PAGE LENGTH field specifies the length of the SUPPORTED PAGE LIST. If the ALLOCATION LENGTH is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The SUPPORTED PAGE LIST field shall contain a list of all vital product data page codes (see Section 8.6) implemented for the target or logical unit in ascending order beginning with page code 00h.

8.6.5.5 Unit Serial Number page

This page provides a product serial number for the target or logical unit. See Table 100 following.

Table 100. Unit Serial Number page (80h)

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (80h)[2]							
2	Reserved							
3	PAGE LENGTH [3]							
4 : n	Product Serial Number [4]							

[1] The PERIPHERAL QUALIFIER field and the PERIPHERAL DEVICE TYPE field are defined in Table 77, Note [1].

[2] PAGE CODE field.

[3] The PAGE LENGTH field specifies the length in bytes of the product serial number page. Older products that only support the Product Serial Number parameter will have a page length of 08h, while newer products that support both parameters will have a page length of 14h. If the ALLOCATION LENGTH is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The Product Serial Number field contains ASCII data that is vendor-assigned serial number. The least significant ASCII character of the serial number shall appear as the last byte in the Data-In Buffer. If the product serial number is not available, the target shall return ASCII spaces (20h) in this field.

8.6.6 Firmware Numbers page (C0h)

Table 101 applies to drives that do not supply a servo RAM Release number and date and a ROM Release date. Table 102 applies to model families other than those covered by Table 101.

Table 101. Firmware Numbers page (C0h)

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]		PERIPHERAL DEVICE TYPE [1]					
1	PAGE CODE (C0h) [2]							
2	0	0	0	0	0	0	0	0
3	PAGE LENGTH (18h) [3]							
4 : 11	SCSI FIRMWARE RELEASE NUMBER [4]							
12 : 19	ASCII space characters							
20 : 27	Servo ROM Release Number [4] [5]							

[1] The PERIPHERAL QUALIFIER field and PERIPHERAL DEVICE TYPE field value of 00h indicates a direct-access device (magnetic disc) is connected to this logical unit.

[2] PAGE CODE (C0h) provides the Firmware Release numbers for the drive (see Table 102).

[3] The PAGE LENGTH field specifies the length of the product firmware numbers. If the ALLOCATION LENGTH is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The SCSI FIRMWARE RELEASE NUMBER field contains ASCII data. The least significant ASCII character of the drive firmware number shall appear as the last byte of a successful data transfer.

[5] For drives that return only 28 bytes. For drives that return more than 28 bytes, see Table 102.

Table 102. Firmware Numbers page (C0h) (Applies to model families [5][9] other than those covered by Table 101)

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]		PERIPHERAL DEVICE TYPE [1]					
1	PAGE CODE (C0h) [2]							
2	0	0	0	0	0	0	0	0
3	PAGE LENGTH [3]							
4 : 11	SCSI FIRMWARE RELEASE NUMBER [4]							
12 : 19	SERVO FIRMWARE RELEASE NUMBER [4]							
20 : 27	SAP BLOCK POINT NUMBERS (MAJOR/MINOR) [4]							
28 : 31	SERVO FIRMWARE RELEASE DATE [5]							
32 : 35	SERVO ROM RELEASE DATE							
36 : 43	SAP FIRMWARE RELEASE NUMBER [6] [9]							
44 : 47	SAP FIRMWARE RELEASE DATE [6] [9]							
48 : 51	SAP FIRMWARE RELEASE YEAR [6] [9]							
52 : 55	SAP MANUFACTURING KEY [7] [9]							
56 : 59	SERVO FIRMWARE PRODUCT FAMILY AND PRODUCT FAMILY MEMBER IDs [8] [9]							

[1] The PERIPHERAL QUALIFIER field and PERIPHERAL DEVICE TYPE field value of 00h indicates a direct-access device (magnetic disc) is connected to this logical unit.

[2] PAGE CODE (C0h) provides the Firmware Release numbers for the drive.

[3] The PAGE LENGTH field specifies the length of the product firmware numbers. If the ALLOCATION LENGTH is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The SCSI FIRMWARE RELEASE NUMBER fields contain ASCII data. The least significant ASCII character of the Drive firmware number shall appear as the last byte of a successful data transfer.

[5] Applies to model families that return 36 bytes rather than 28. Newer drive models return 60 bytes. Servo firmware RAM and ROM Release dates are added in some models. See “Vital Product Data pages supported” table in the Product Manual, Volume 1, that applies to the drive in question.

[6] Servo Adaptive Parameters (SAP) are created independently from the servo firmware. Thus, they have their own release information. The SAP Firmware Release Number field contains ASCII data.

[7] The SAP MANUFACTURING KEY field contains a binary code used by manufacturing to identify the SAP.

[8] The high order word of the SERVO FIRMWARE PRODUCT FAMILY and PRODUCT FAMILY MEMBER IDs field contains a binary code which corresponds to the Servo Firmware Family ID. The low order word of this field contains a binary code which corresponds to the Product Family Member ID.

[9] Applies to model families that return 59 bytes rather than 35 or 28. Servo firmware RAM and ROM Release dates are added, plus SAP firmware. See “Vital Product Data pages supported” table in the Product Manual, Volume 1, that applies to the drive in question.

8.6.7 Date Code page (C1h)

Table 103. ETF Log Date Code page (C1h)

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]		PERIPHERAL DEVICE TYPE [1]					
1	PAGE CODE (C1h) [2]							
2	0	0	0	0	0	0	0	0
3	PAGE LENGTH (10h) [3]							
4	(MSB)							
:	ETF LOG DATE IN ASCII [4]							
11								(LSB)
12	(MSB)							
:	COMPILE DATE CODE [5]							
19								(LSB)

[1] The PERIPHERAL QUALIFIER field and PERIPHERAL DEVICE TYPE field value of 00h indicates a direct-access device (magnetic disc) is connected to this logical unit.

[2] PAGE CODE (C1h) provides the date code from the drive defect list.

[3] The PAGE LENGTH field specifies the length of the product date code. If the ALLOCATION LENGTH is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The ETF Log date code field contains ASCII data. The data is stored in the format MMDDYYYY. Leading ASCII zero characters are added to single-digit months or days.

[5] The COMPILE DATE CODE field contains 8 ASCII bytes of data for a date of the form MMDDYYYY.

8.6.8 Jumper Settings page (C2h)

Table 104. Jumper Settings page (C2h)

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (C2h) [2]							
2	0	0	0	0	0	0	0	0
3	PAGE LENGTH (02h) [3]							
4	DS [4]	MS [5]	WP [6]	PE [7]	DRIVE ID [8]			
5	Reserved							TERM ENABLE [9] [10]

[1] The PERIPHERAL QUALIFIER field and PERIPHERAL DEVICE TYPE field value of 00h indicates a direct-access device (magnetic disc) is connected to this logical unit.

[2] PAGE CODE (C2h) provides all the jumper settings for the drive. The bits in byte 4 indicate which jumpers are on.

[3] The PAGE LENGTH field specifies the length of the Jumper Setting page. If the ALLOCATION LENGTH is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The Delayed Motor Start (DS) bit when set to 1 indicates that this jumper is on.

[5] The Motor Start (MS) bit when set to 1 indicates that the jumper is on.

[6] The Write Protect (WP) bit when set to 1 indicates that the write protect jumper is on.

[7] On some drives, Parity Enable (PE) bit when set to 1 indicates that SCSI parity error checking jumper is on. Other drives have a PD (PARITY DISABLE) jumper, where Parity is enabled when the jumper is off. See individual drive's Product Manual, Volume 1, or Installation Guide for information on how the drive of interest is set up.

[8] The Drive ID is shown below in the table. Bit 3 is the most significant bit and bit 0 is the least significant bit.

Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Drive ID**
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

[9] Terminator Enable (TE). Not used on all drives. See individual drive’s Product Manual, Volume 1, for applicability.

[10] It is not presently possible to return information on the Terminator Power (TP) jumpers.

8.6.9 Device Behavior page (C3h)

The Device Behavior page (Table 105) will be used by the regression tests to determine what behavior should be expected from a particular firmware package.

Table 105. Device Behavior page (C3h)

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (C3h) [2]							
2	Reserved							
3	PAGE LENGTH [3]							
4	VERSION NUMBER [4]							
5	BEHAVIOR CODE [5]							
6	BEHAVIOR CODE VERSION NUMBER [5]							
7 : 22	ASCII FAMILY NUMBER (16 bytes) [6]							
23	NUMBER OF INTERLEAVES [7]							
24	DEFAULT NUMBER OF CACHE SEGMENTS [8]							
25+	Feature Flags and Additional Byte Fields will go here but are undefined at this time.							

[1] The PERIPHERAL QUALIFIER field and PERIPHERAL DEVICE TYPE field value of 00h indicates a direct-access device (magnetic disc) is connected to this logical unit.

[2] PAGE CODE field.

[3] PAGE LENGTH defines the length of the Device Behavior information in bytes. If the ALLOCATION LENGTH is too small to transfer all of the page, the page length shall not be adjusted to reflect the truncation.

[4] The VERSION NUMBER is a one-byte short form notation for the 24-byte assignment in the Firmware Numbers page. Version Numbers are registered by Engineering services.

[5] The BEHAVIOR CODE and BEHAVIOR CODE VERSION NUMBER are jointly assigned by the Firmware Engineering Managers of all SCSI design locations.

[6] The ASCII FAMILY NUMBER field contains the drive model number. The data is left-aligned within this field.

[7] The NUMBER OF INTERLEAVES field indicates the number of data interleaves used by the ECC correction algorithm.

[8] The DEFAULT NUMBER OF CACHE SEGMENTS field gives the number of segments into which the host requests the drive divide the cache.

8.7 LOCK UNLOCK CACHE (10) command (36h)

The LOCK UNLOCK CACHE (10) command (see Table 106) requests that the target disallow or allow logical blocks within the specified range to be removed from the cache memory by the target's cache replacement algorithm. Locked logical blocks may be written to the medium when modified, but a copy of the modified logical block shall remain in the cache memory.

Table 106. LOCK UNLOCK CACHE (10) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (36h)							
1	Reserved [1]						LOCK [2]	RELADR [3]
2	(MSB)	LOGICAL BLOCK ADDRESS [4]						
3								
4								
5								(LSB)
6	Reserved							
7	(MSB)	NUMBER OF BLOCKS [5]						
8								(LSB)
9	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A LOCK bit of zero indicates that all logical blocks in the specified range that are currently locked into the cache memory shall be unlocked, but may not be removed. A LOCK bit of one indicates that any logical block in the specified range that is currently present in the cache memory shall be locked into cache memory. Only logical blocks that are already present in the cache memory are actually locked.

[3] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[4] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [3]).

[5] The NUMBER OF BLOCKS field specifies the total number of contiguous logical blocks within the range. A NUMBER OF BLOCKS field of zero indicates that all remaining logical blocks on the block device shall be within the range.

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

Multiple locks may be in effect from more than one initiator. Locks from different initiators may overlap. An unlock of an overlapped area does not release the lock of another initiator.

8.8 LOCK UNLOCK CACHE (16) command (92h)

The LOCK UNLOCK CACHE (16) command (see Table 107) requests that the target disallow or allow logical blocks within the specified range to be removed from the cache memory by the target's cache replacement algorithm. Locked logical blocks may be written to the medium when modified, but a copy of the modified logical block shall remain in the cache memory.

Table 107. LOCK UNLOCK CACHE (16) command

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (92h)							
1	Reserved [1]						LOCK [2]	RELADR [3]
2	(MSB)	LOGICAL BLOCK ADDRESS [4]						
3								
4								
5								
6								
7								
8								
9								(LSB)
10	(MSB)	NUMBER OF BLOCKS [5]						
11								
12								
13								(LSB)
14	Reserved							
15	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A LOCK bit of zero indicates that all logical blocks in the specified range that are currently locked into the cache memory shall be unlocked, but may not be removed. A LOCK bit of one indicates that any logical block in the specified range that is currently present in the cache memory shall be locked into cache memory. Only logical blocks that are already present in the cache memory are actually locked.

[3] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block

of the range of logical blocks to be operated on by this command.

[4] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [3]).

[5] The NUMBER OF BLOCKS field specifies the total number of contiguous logical blocks within the range. A NUMBER OF BLOCKS field of zero indicates that all remaining logical blocks on the block device shall be within the range.

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

Multiple locks may be in effect from more than one initiator. Locks from different initiators may overlap. An unlock of an overlapped area does not release the lock of another initiator.

8.9 LOG SELECT command (4Ch)

The LOG SELECT command provides a means for an initiator to manage statistical information maintained by the device about the device or its logical units. This information is logged within the drive and can be sent to the initiator in response to a LOG SENSE command from the initiator. The LOG SELECT command format is shown in Table 108. targets that implement the LOG SELECT command shall also implement the LOG SENSE command. Structures in the form of log parameters within log pages are defined as a way to manage the log data. The LOG SELECT command provides for sending zero or more log pages via the Data Out Buffer in the Log Page Format of Table 112. These Log Pages contain parameters that command the drive to change selected threshold, or cumulative values of any or all drive logs. This section defines the format

The following tables in this section, and those in Section 8.9 that apply for the LOG SELECT command, show functions that command the drive to perform or enable for performance, control parameter bits the drive shall set/reset/save, log counts that shall be kept, etc. The list of Log pages supported is given in Table 115.

For the LOG SENSE command (see Section 8.10), the tables there apply as indicators of functions the drive reports back to the host that it is enabled to perform, control parameter bits that are set/reset/saved, log counts that are being kept, etc. Although Section 8.10 describes the LOG SENSE command, the tables of that section that apply to the LOG SELECT command are not repeated here in this section. Also, though the language of the descriptions in Section 8.10 is for the LOG SENSE case, they apply to the LOG SELECT case similarly, unless noted otherwise.

The drives represented by this Interface Manual do not support keeping independent sets of log parameters (one set for each initiator in the system). If at some point log parameters are changed (by a LOG SELECT command) that affect initiators other than the initiator that sent the LOG SELECT command, the drive generates a UNIT ATTENTION condition for those other initiators, but not for the one that issued the LOG SELECT command. When the other initiators at a future time connect to the drive, the first command attempted will not execute and a CHECK CONDITION status will be issued by the drive. A REQUEST SENSE command would normally follow and a UNIT ATTENTION condition sense code would be returned to these other initiators with an additional sense code of Log Parameters Changed (one by one as they connect to the drive). See Section 7.6.6, UNIT ATTENTION condition.

Table 108. LOG SELECT command (4Ch)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (4Ch)							
1	Reserved [1]						PCR [2]	SP [3]
2	PC [4]		Reserved					
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	(MSB)	PARAMETER LIST LENGTH [5]						
8								(LSB)
9	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Parameter Code Reset (PCR) bit of one and a PARAMETER LIST LENGTH of zero causes all implemented parameters to be set to the drive-defined default values (most likely zero). If the PCR bit is one and the PARAMETER LIST LENGTH is greater than zero, the command is terminated with a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to Invalid Field in CDB. A PCR bit of zero specifies that the log parameters shall not be reset.

[3] A Save Parameters (SP) bit of one indicates that after performing the specified LOG SELECT operation the drive shall save to non-volatile memory all LOG SELECT parameters identified as savable by the DS bit in the Log parameter sections of the Log page (see Tables 112 and 113). An SP bit of zero specifies that parameters shall not be saved. Log Parameters are also saved after each thermal calibration if the TSD bit in the Log Parameter pages (see Table 113) is zero.

It is not an error to set the SP bit to one and to set the DS bit of a LOG PARAMETER to one. In this case, the parameter value for that LOG PARAMETER is not saved.

[4] The Page Control (PC) field defines the type of LOG PARAMETER the initiator selects to change with the LOG SELECT command. The PC field is defined in Table 109.

Table 109. Page CONTROL field

PC Field Value	Type of Log Parameter
00b	Log Current Threshold Values
01b	Log Current Cumulative Values
10b	Log Default Threshold Values
11b	Log Default Cumulative Values

The drive only updates the cumulative values to reflect the number of events experienced by the drive, but the initiator can set the threshold or cumulative log (00 or 01) parameter values using the LOG SELECT command with the PC field set as applicable.

The drive sets the current log values to default values in response to a LOG SELECT command with the PARAMETER LIST LENGTH set to zero and the PC field set to the applicable value (10 or 11) per Table 109.

If a SCSI initiator port attempts to change a current threshold value that is not available or not implemented for that LOG PARAMETER, the drive shall terminate the LOG SELECT command with a CHECK CONDITION status, the sense key set to ILLEGAL REQUEST and an additional sense code set to Invalid Field In Parameter List. The saving of current threshold parameters and the criteria for the current threshold being met are controlled by bits in the Parameter Control Byte (PCB) (byte 2 of each of the LOG PARAMETER pages).

[5] The PARAMETER LIST LENGTH specifies the length in bytes of the parameter list that shall be transferred from the initiator to the drive during the DATA OUT phase. A PARAMETER LIST LENGTH of zero indicates that no pages shall be transferred. This condition shall not be considered an error.

If the initiator sends page codes or parameter codes within the parameter list that are reserved or not implemented by the drive, the drive shall terminate the LOG SELECT command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code set to Invalid Field In Parameter List.

If a PARAMETER LIST LENGTH results in the truncation of any log parameter, the drive shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code set to Invalid Field In CDB.

In the DATA OUT phase of the LOG SELECT command, the initiator may send none, one, or more data pages, each of which is in the format specified in Table 112 and which contain control information pertaining to the management and reporting of various drive log parameters. If multiple pages are sent out following the command CDB, they must be sent in ascending page code value order. Also, Log Parameters in each Log Page must be sent in Log Parameter Code ascending order (see note [4] in Table 112). The

PAGE CODE (byte 0) specifies which log the page pertains to. The PAGE CODE assignments are listed in Table 115. Initiator cannot send page zero.

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

8.10 LOG SENSE command (4Dh)

The LOG SENSE command provides a means for a SCSI initiator port to retrieve statistical information maintained by the drive about the drive operation. It is a complementary command to the LOG SELECT command. This information is stored in logs (counters) in the drive and is sent to the initiator in the DATA IN phase of the LOG SENSE command. The LOG SENSE command format that the initiator sends is shown in Table 110. The format of the data pages sent back by the drive in the DATA IN phase is shown in Table 111.

Table 110. LOG SENSE command (4Dh)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (4Dh)							
1	Reserved						PPC [1]	SP [2]
2	PC [3]		PAGE CODE [4]					
3	Reserved							
4	Reserved							
5	(MSB)			PARAMETER POINTER [5]				
6								(LSB)
7	(MSB)			ALLOCATION LENGTH [6]				
8								(LSB)
9	CONTROL [7]							

[1] A Parameter Pointer Control (PPC) bit of one indicates that the drive shall return a log page with only those log parameters that have changed since the last LOG SELECT or LOG SENSE command. The drive returns log parameter codes according to (in ascending numerical order) the parameter code specified in the PARAMETER POINTER (bytes 5 & 6, see note [5]).

A PPC bit of zero indicates that all of the log parameter data requested from the drive, whether changed or not, is sent and it begins with the log specified by the parameter code given in the PARAMETER POINTER field and returns the number of bytes specified by the ALLOCATION LENGTH field, in ascending order of parameter codes starting at the parameter code given in the PARAMETER POINTER field. A PPC bit of zero and a PARAMETER POINTER field of zero causes all available log parameters for the specified log page to be returned to the initiator subject to the ALLOCATION LENGTH.

[2] If the Save Parameters (SP) bit is zero, the drive performs the specified LOG SENSE command and does not save any log parameters to non-volatile memory. If SP bit is one, the drive first saves parameters identified as savable (by the DS bit in Table 113) to a nonvolatile location, and then performs the rest of the LOG SENSE command.

[3] The Page Control (PC) field defines the type of drive log parameter the initiator requests the drive to send back on the DATA IN phase. See Table 109 for definition of this field. The parameter values returned on the DATA IN phase are from one of the following:

a. The specified parameter values in the log counters as of the last update (updated by LOG SELECT command, LOG SENSE command or done automatically by the drive for cumulative values).

b. The saved values if an update has not occurred since last power-on, hard reset condition, or Bus Device Reset message (assumes saved values are available).

c. If saved values are not available, the default values are sent if an update has not occurred since the last power-on, hard reset condition, or Bus Reset message.

[4] The PAGE CODE field specifies the page of data requested by the command (see Table 115) (This page is not defined for the LOG SELECT command). If a page code is sent that is not implemented, the drive terminates the command with CHECK CONDITION status. The sense key is set to ILLEGAL REQUEST with the additional sense code set to Invalid Field In CDB.

Table 111 (PAGE CODE 00h) returns the list of log pages that the drive supports.

Table 111. Supported Log pages

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved		PAGE CODE (00h)					
1	Reserved							
2	(MSB)	PAGE LENGTH (n–3) [a]						
3								(LSB)
4 : n	SUPPORTED PAGE LIST [b]							

[a] The PAGE LENGTH field specifies the length in bytes of the following SUPPORTED PAGE LIST.

[b] The SUPPORTED PAGE LIST field shall contain a list of all log page codes implemented by the target in ascending order beginning with PAGE CODE 00h.

[5] The PARAMETER POINTER field contains a PARAMETER CODE that specifies that log parameter data be returned to the initiator by the DATA IN phase starting with the Parameter Pointer code log parameter data and continuing to the maximum ALLOCATION LENGTH or to (and including) log parameter data of the maximum parameter code supported by the drive, whichever is less. If the value of the PARAMETER POINTER field is larger than the largest available parameter code that can be returned by the drive on the specified page, the drive terminates the command with a CHECK CONDITION status. The sense key is set to ILLEGAL REQUEST and the additional sense code is set to Invalid Field In CDB.

[6] The ALLOCATION LENGTH field informs the drive of the amount of space available for returning log parameter data. The initiator can retrieve the rest of the log page information by setting the PARAMETER POINTER to the last returned parameter code and reissuing the LOG SENSE command. This process may be repeated as necessary to retrieve all the available information.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

The LOG SENSE command and its DATA IN phase use the format of Table 112. Explanations pertaining to the table apply generally to both LOG SELECT and LOG SENSE commands, with differences noted. However, only one log page at a time is returned with each LOG SENSE command (see Section 8.10). This table is not given in Section 8.9, though it applies.

Table 112. Log Page format

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE [1]							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3) [2]						
3								(LSB)

Log Parameter Structure(s) [3]

4 to x+3	LOG PARAMETER (First) [4] (Length X bytes)
:	: [4]
n-Y+1 to n	LOG PARAMETER (Last) [4] (Length Y bytes)

[1] PAGE CODE. Refer to Table 115.

[2] The PAGE LENGTH field gives the total number of bytes of LOG PARAMETER structures that follow these first four control block bytes. If the initiator sends a PAGE LENGTH that results in the truncation of any parameter, the target shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST with the additional sense code set to Invalid Field In Parameter List.

[3] Most log pages contain one or more special data structures called Log Parameters. Log Parameters may be data counters that record a count of a particular event (or events) or list parameters (strings) that contain a description of a particular event. List parameters are not currently supported by the drives represented by this manual.

[4] Each LOG PARAMETER structure begins with a four-byte parameter header followed by one or more bytes of parameter value data. LOG PARAMETER structures are in the format given in Table 113.

Table 113. Log Parameters

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PARAMETER CODE [1]						
1								(LSB)
2	DU [2]	DS [3]	TSD [4]	ETC [5]	TMC [6]		LBIN [7]	LP [8]
3	PARAMETER LENGTH (n-3 bytes) [9]							
4 : n	– – – –	PARAMETER VALUE [10]						– – – –

[1] The PARAMETER CODE field identifies the specific parameter that is being transferred with the Log Page. These codes are listed and explained in the individual page code descriptions following Table 115.

Byte 2 is referred to as the PARAMETER CONTROL byte. The meanings of the various fields are discussed in notes [2] through [8]. For a LOG SELECT command, these bits perform a control function, but on a LOG SENSE command, they only report the drive settings of these bits in this same format on the Data In part of the LOG SENSE command.

[2] Disable Update (DU). For the LOG SELECT command, this applies only to the cumulative LOG PARAMETER values (indicated by 01 in the PC field of the LOG SELECT and LOG SENSE command descriptor block). The drive can set this bit to one or zero also (see note [8] following).

The DU flag bit is defined as follows:

a. A zero value indicates that the drive shall update the LOG PARAMETER value to reflect all events that should be logged by that parameter.

b. A one value indicates that the drive shall not update the LOG PARAMETER value except in response to a LOG SELECT command that specifies a new value for the parameter.

The DU flag is set to one when the current cumulative value of the parameter counter it controls reaches its maximum value (see note [9]). Upon reaching this maximum value, the data counter does not wrap around and start over at zero. Incrementing of other counters within the same log pages ceases. Counters do not restart automatically if the overflowed counter is re-initialized. If the data counter reaches its maximum value during the execution of a command, the drive completes the command. Drive counter updates are performed in the background. This means a counter may overflow long after a command has completed, so the drive must treat this condition as a UNIT ATTENTION with the additional sense code set to Log Counter at max for all initiators if RLEC=1 (Report Log Exception Condition bit of the Control Mode Page 0Ah).

Since the drive uses volatile memory to hold cumulative values, they will be lost when a power cycle occurs. Unless the initiator commands the drive to save them to non-volatile memory using a LOG SELECT or LOG SENSE command with the SP bit set to one.

The DU bit is not defined for threshold values (indicated by the PC field of the LOG SENSE command descriptor block) nor for list parameters (indicated by the LP bit). The drive ignores the value of DU bits in a LOG SELECT command applicable to threshold values or list parameters.

[3] If Disable Save (DS) is zero, it indicates that the drive supports LOG SELECT and LOG SENSE data saving for that LOG PARAMETER. The drive saves the current cumulative and the current threshold parameter values in response to a LOG SELECT or LOG SENSE command with an SP bit of one. A DS bit of one

indicates that the drive does not support saving that LOG PARAMETER in response to a LOG SELECT or LOG SENSE command with a SP bit of one.

[4] Target Save Disable (TSD). A zero indicates that the drive shall save frequently enough to insure statistical significance. The drive's method is to save after each thermal calibration, which is typically (may not always be) once every ten minutes. A one bit indicates that the drive does not use its save method.

[5] Enable Threshold Comparison (ETC). A one indicates that a comparison to the threshold value is performed whenever the cumulative value is updated. A bit of zero indicates the comparison is not performed. The value of the ETC bit is the same for both the threshold and cumulative parameters.

[6] The Threshold Met Criteria (TMC) field defines the basis for comparison of the cumulative and threshold values. See Table 114 for meanings of values in this field. The TMC field is only valid when the ETC bit is one.

Table 114. Threshold Met Criteria

Code	Basis for Comparison
00b	Notify of every update of cumulative value
01b*	Cumulative value equal to threshold value
10b*	Cumulative value not equal threshold value
11b*	Cumulative value greater than threshold value

*Comparison made at every update of cumulative value.

If the ETC bit is one* and the result of the comparison is true, a UNIT ATTENTION Condition is generated for all initiators. When reporting the UNIT ATTENTION Condition, the drive sets the sense key to UNIT ATTENTION, and the additional sense code to Threshold Condition Met.

*The RLEC bit (Report Log Exception Condition) in Mode page 0AH (Table 162) must also be one.

[7] The LBIN bit is only valid if the LP bit is set to one. If the LP bit is one and the LBIN bit is zero then the list parameter is ASCII data. If the LP bit is one and the LBIN bit is one then the list parameter is a list of binary information.

[8] List Parameter (LP). Zero indicates the parameter is a data counter. One indicates that the parameter is a list parameter. This bit only has meaning for the LOG SENSE command Data In pages.

Data counters are associated with one or more events. The data counter is updated whenever one of these events occurs by incrementing the counter value, provided the DU bit is zero. See note [2] for this table.

An LP bit of one indicates that the parameter is a list parameter. List parameters are not counters and thus the ETC and TMC fields shall be set to zero. A list parameter is a string of ASCII graphic codes (i.e., code values 20h thru 73h). List parameters are not supported by the drive at this time.

[9] PARAMETER LENGTH. This field specifies the length in bytes of the parameter that follows. If the initiator sends a PARAMETER LENGTH value that results in the truncation of the PARAMETER VALUE, the drive terminates the command with a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST with the additional sense code set to Invalid Field In Parameter List.

[10] PARAMETER VALUE. This field uses one, two, four, or eight bytes to transmit an unsigned counter value. The initiator sends counts to set values into counters in the drive, and the drive returns counter values to the initiator. The initiator is responsible to issue a LOG SENSE command to learn the PARAMETER LENGTH the target has selected.

When any counter in a log page reaches its maximum value, the drive ceases incrementing all counters in that log page. If the RLEC bit of the Control mode page is one, then the drive reports the exception condition as described in note [2] of this table.

The page code assignments for the log pages are listed in Table 115. Detailed descriptions follow the table.

Table 115. Log Page codes

Page Code	Description	Section	Page
0Fh	Initiator page	8.10.1	209
01h	Buffer Over-run/Under-run page	8.10.2	210
37h	Cache Statistics page	8.10.3	211
03h	Error Counter page (READ)	8.10.4	212
04h	Error Counter page (READ REVERSE)	8.10.4	212
05h	Error Counter page (Verify)	8.10.4	212
02h	Error Counter page (Write)	8.10.4	212
3Eh	Factory Log page	8.10.5	213
2Fh	Informational Exceptions page	8.10.6	213
0Bh	Last *n* Deferred Errors or Asynchronous Events page	8.10.7	215
07h	Last *n* Error Events page	8.10.8	215
06h	Non-Medium Error page	8.10.10	217
18h	Protocol Specific Port page	8.10.9	215
10h	Self-Test Results page	8.10.11	218
0Eh	Start-Stop Cycle Counter page	8.10.12	221
00h	Supported Log pages	8.10.13	224
0Dh	Temperature page	8.10.14	225
08h - 0A	Reserved		
0Ch	Reserved		
11h - 2Fh	Reserved (18h used above)		
3Fh	Reserved		
30h - 3Dh	vendor-specific (37h used above)		

8.10.1 Initiator page (0Fh)

The Initiator page (Table 116) provides a place for initiators to store system information. The page code for the Initiator page is 0Fh.

Table 116. Initiator page

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (0Fh) [1]							
1	Reserved							
2	(MSB)	PAGE LENGTH [1]						
3								(LSB)

Initiator log parameters [2]

4	FIRST APPLICATION LOG PARAMETER							
:	:							
	LAST APPLICATION LOG PARAMETER							
n								

[1] The PAGE CODE and PAGE LENGTH fields are defined in Notes [1] and [2] of Table 112.

[2] Parameter codes 0000h through 0FFFFh are for general usage initiator data. The intended use for this information is to aid in describing the system configuration and system problems, but the exact definition of the data is initiator specific. The general usage initiator data parameters all have the format shown in Table 117.

Table 117: General usage initiator parameter data

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PARAMETER CODE [a]						
1								(LSB)
2	DU [d]	DS [d]	TSD [d]	ETC [d]	TMC [d]		LBIN [d]	LP [d]
3	PARAMETER LENGTH (FCh) [b]							
4	GENERAL USAGE PARAMETER BYTES [c]							
255								

[a] For general usage initiator data, the value in the PARAMETER CODE field shall be between 0000h and 0FFFh. The first supported general usage initiator parameter code shall be 0000h and additional supported parameters shall be sequentially numbered. If any general usage parameter codes are implemented, the device shall support at least 64 general usage parameter descriptors and they shall be parameter codes 0000h through 003Fh.

In the initiator page, parameter codes 1000h through FFFFh are reserved.

[b] For the general usage initiator parameter, the PARAMETER LENGTH value for each parameter shall be FCh.

[c] The values stored in the GENERAL USAGE PARAMETER BYTES represent data sent to the target in a previous LOG SELECT command. If a previous LOG SELECT command has not occurred, the data is vendor-specific.

[d] The state of the LOG PARAMETER control bits ([4] through[10]) for parameters 0000h through 0FFFh is specified in Table 118.

Table 118: Parameter control bits for general usage parameters (0000h through 0FFFh)

Bit	Value	Description
DU	1	Value provided by applications client
DS	0	target supports saving of parameter
TSD	0	target manages saving of parameter
ETC	0	No threshold comparison is made on this value
TMC	xxx	Ignored when ETC is 0
LBIN	1	The parameter is in binary format
LP	1	The parameter is a list parameter

8.10.2 Buffer Over-run/Under-run page (01h)

The Buffer Over-run/Under-run page (page code 01h) defines 24 data counters that may be used to record the number of buffer over-runs or under-runs for the logical unit. a SCSI target port that implements this page may implement one or more of the defined data counters.

A buffer over-run or under-run may occur when a SCSI initiator port does not transmit data to or from the target's buffer fast enough to keep up with reading or writing the media. The cause of this problem is protocol-specific. A buffer over-run condition may occur during a Read operation when a buffer full condition prevents continued transfer of data from the media to the buffer. A buffer under-run condition may occur during a Write operation when a buffer empty condition prevents continued transfer of data to the media from the buffer. Most devices incur a delay at this point while the media is repositioned.

The PARAMETER CODE field for buffer over-run/under-run counters is a 16-bit value comprised of eight reserved bits, a three-bit COUNT BASIS field (see Table 120), a four-bit CAUSE field (see Table 121), and a one-bit TYPE field. These are concatenated to determine the value of the parameter code for that LOG PARAMETER. For example, a counter for parameter code value of 0023h specifies a count basis of 001b; a cause of 0001b; and a type of 1b; this counter is incremented once per command that experiences an over-run due to the SCSI bus being busy.

Table 119 defines the PARAMETER CODE field for the buffer over-run/under-run counters.

Table 119. Parameter code field for buffer over-run/under-run counters

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	COUNT BASIS [1]			CAUSE [2]				TYPE [3]

[1] The COUNT BASIS field defines the criteria for incrementing the counter.

Note. The per unit of time count basis is device type specific. Direct-access devices typically use a latency period (i.e., one revolution of the medium) as the unit of time.

Table 120: **Count basis definition**

Count basis	Description
000b	Undefined
001b	Per command
010b	Per failed reconnect
011b	Per unit of time
100b - 111b	Reserved

[2] The CAUSE field indicates the reason that the over-run occurred.

Table 121: Cause field definition

Cause	Description
0h	Undefined
1h	Bus busy
2h	Transfer rate to slow
3h - Fh	Reserved

[3] The TYPE field indicates whether the counter records under-runs or over-runs. A value of zero specifies a buffer under-run condition and a value of one specifies a buffer over-run condition.

The counters contain the total number of times buffer over-run or under-run conditions have occurred since the last time the counter was cleared. The counter shall be incremented for each occurrence of an under-run or over-run condition and may be incremented more than once for multiple occurrences during the execution of a single command.

8.10.3 Cache Statistics page (37h)

Log Page code 37h specifies Cache Statistics page. The page format is shown in Table 122.

Table 122. Cache Statistics page (37h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	PARAMETER CODE [1]							

[1] Parameter codes 00h through 04h are described following:

00h This parameter code represents the number of logical blocks that have been sent to a SCSI initiator port.

01h This parameter code represents the number of logical blocks that have been received from a SCSI initiator port.

02h This parameter code represents the number of logical blocks read from the cache memory that have been sent to a SCSI initiator port.

03h This parameter code represents the number of READ and WRITE commands that had data lengths equal or less than the current segment size.

04h This parameter code represents the number of READ and WRITE commands that had data lengths greater that the current segment size.

8.10.4 Error Counter pages (WRITE, READ, READ REVERSE and VERIFY, 02h, 03h, 04h, and 05h)

This clause defines the optional error counter pages for WRITE errors (page code 02h), READ errors (page code 03h), READ REVERSE errors (page code 04h) and VERIFY errors (page code 05h). The log page format is defined near the beginning of 8.10.1. A page may return one or more log parameters that record events defined by the parameter codes. Table 123 defines the parameter codes for the error counter pages. Support of each log parameter is optional.

Table 123. Error Counter pages (WRITE, READ, READ REVERSE, and VERIFY) parameter code field

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	PARAMETER CODE [1]							

[1] PARAMETER CODE values 0000h through 0006h specify six counters each for WRITE, READ, READ REVERSE and VERIFY errors (18 counters). A description of the type (category of error) counters specified by codes 00h through 06h are described following.

0000h Error Corrected Without Substantial Delay. An error correction was applied to get perfect data (a.k.a., ECC on-the-fly). “Without Substantial Delay” means the correction did not postpone reading of later sectors (e.g., a revolution was not lost). The counter is incremented once for each logical block that requires correction. Two different blocks corrected during the same command are counted as two events.

0001h Error Corrected With Possible Delays. An error code or algorithm (e.g., ECC, checksum) is applied in order to get perfect data with substantial delay. “With possible delay” means the correction took longer than a sector time so that reading/writing of subsequent sectors was delayed (e.g, a lost revolution). The counter is incremented once for each logical block that requires correction. A block with a double error that is correctable counts as one event and two different blocks corrected during the same command count as two events.

0002h Total (e.g., re-writes or re-reads). This parameter code specifies the counter counting the number of errors that are corrected by applying retries. This counts errors recovered, not the number of retries. If five retries were required to recover one block of data, the counter increments by one, not five. The counter is incremented once for each logical block that is recovered using retries. If an error is not recoverable while applying retries and is recovered by ECC, it isn’t counted by this counter; it will be counted by the counter specified by parameter code 01h–Error Corrected With Possible Delay.

0003h Total Error Corrected. This counter counts the total of parameter code errors 00h, 01h, and 02h. There is to be no “double counting” of data errors among these three counters. The sum of all correctable errors can be reached by adding parameter code 01h and 02h errors, not by using this total.

0004h Total Times Correction Algorithm Processed. This parameter code specifies the counter that counts the total number of retries, or “times the retry algorithm is invoked.” If after five attempts a counter 02h type error is recovered, then five is added to this counter. If three retries are required to get a stable ECC syndrome before a counter 01h type error is corrected, then those three retries are also counted here. The number of retries applied to unsuccessfully recover an error (counter 06h type error) are also counted by this counter.

0005h Total Bytes Processed. This parameter code specifies the counter that counts the total number of bytes either successfully or unsuccessfully read, written, or verified (depending on the log page) from the drive. If a transfer terminates early because of an unrecoverable error, only the logical blocks up to and including the one with the unrecoverable error are counted.

0006h Total Uncorrected Errors. This parameter code specifies the counter that contains the total number of blocks for which an uncorrected data error has occurred.

8.10.5 Factory Log page (3Eh)

Log page code 3Eh specifies factory status parameters (see Table 124).

Table 124. Factory Log page (3Eh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	PARAMETER CODE [1] [2]							

[1] PARAMETER CODE 0000h–Power-on Time. This parameter code represents the number of drive power-on minutes. Currently the Power-on Time parameter (0000h) is the only parameter in this Log Page that is visible to OEM/customers.

[2] PARAMETER CODE 08h. This parameter reports the time, in minutes, to the next scheduled interrupt for a S.M.A.R.T. measurement (see Section 9.1).

8.10.6 Informational Exceptions log page (2Fh)

The Informational Exceptions log page (see Table 125) provides a place for reporting detail about informational exceptions. The page code for the Informational Exceptions log page is 2Fh.

Table 125. Informational Exceptions log page

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (2Fh) [1]							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3) [1]						
3								(LSB)

Informational exceptions log parameters [2]

4	First informational exceptions log parameter
:	:
n	Last informational exceptions log parameter

[1] The PAGE CODE and PAGE LENGTH fields are defined in Notes [1] and [2] of Table 112.

[2] The INFORMATIONAL EXCEPTIONS LOG provides a list of Informational Exception information, the format of which is given in Table 126.

The informational exceptions general parameter data page has the format shown in Table 126.

Table 126. Informational exceptions general parameter data

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PARAMETER CODE (0000h) [a]						
1								(LSB)
2	DU [b]	DS [b]	TSD [b]	ETC [b]	TMC [b]		LBIN [b]	LP [b]
3	PARAMETER LENGTH (n-3) [c]							
4	INFORMATIONAL EXCEPTION ADDITIONAL SENSE CODE[d]							
5	INFORMATIONAL EXCEPTION ADDITIONAL SENSE CODE QUALIFIER [d]							
6	MOST RECENT TEMPERATURE READING [e]							
7	Vendor specific							
n								

[a] Table 127 defines the parameter codes.

Table 127: Informational exceptions parameter codes

Parameter code	Description
0000h	Informational exceptions general parameter data
0001h - FFFFh	Vendor specific

[b] The values of the log parameter control bits for self test results log parameters is specified in Table 128.

Table 128: Parameter control bits for Informational exceptions log parameter (0000h)

Bit	Value	Description
DU	0	Value provided by target
DS	0	target supports saving of parameter
TSD	0	target manages saving of parameter
ETC	0	No threshold comparison is made on this value
TMC	xxx	Ignored when ETC is 0
LBIN	1	The parameter is in binary format
LP	1	The parameter is a list parameter

[c] The PARAMETER LENGTH field is described in Table 113 on page 206, note [9]. The parameter length shall be at least 04h.

[d] If the INFORMATIONAL EXCEPTION ADDITIONAL SENSE CODE field contains zero, no informational exception condition is pending and contents of the INFORMATIONAL EXCEPTION ADDITIONAL SENSE CODE QUALIFIER field are unspecified. If the INFORMATIONAL EXCEPTION ADDITIONAL SENSE CODE field contains any value other than zero, an informational exception condition exists that has an ADDITIONAL SENSE CODE indicated by INFORMATIONAL EXCEPTION ADDITIONAL SENSE CODE field and an ADDITIONAL SENSE CODE QUALIFIER indicated by the INFORMATIONAL EXCEPTION ADDITIONAL SENSE CODE QUALIFIER field.

[e] The MOST RECENT TEMPERATURE READING field indicates the temperature in degrees Celsius of the SCSI target device at the time the LOG SENSE command is performed. Temperatures equal to or less than zero degrees Celsius shall be indicated by a value of zero. If the target is unable to detect a valid temperature because of a sensor failure or other condition, the value returned shall be FFh. The temperature should be reported with an accuracy of plus or minus three Celsius degrees while the device is operating at a steady state within the environmental limits specified for the device.

8.10.7 Last *n* Deferred Errors or Asynchronous Events page (0Bh)

Log page (0Bh) provides for a number of deferred errors or asynchronous events sense data records using the list parameter format of the log page. The number of these deferred errors or asynchronous events records supported, *n*, is vendor-specific. Each deferred error or asynchronous event record contains SCSI sense data for a deferred error or asynchronous event that has occurred. The parameter code associated with the record indicates the relative time at which the deferred error or asynchronous event occurred. A higher parameter code indicates that the deferred error or asynchronous event occurred later in time.

The content of the parameter value field of each log parameter is the SCSI sense data describing the deferred error.

The fields DU, TSD, ETC, and TMC are reserved and shall be set to zero. The LBIN bit shall be set to one (binary information). The LP bit shall be set to one (list parameter).

8.10.8 Last *n* error events page (07h)

Log page (07h) provides for a number of error-event records using the list parameter format of the log page. The number of these error-event records supported, *n*, is vendor-specific. Each error-event record contains vendor-specific diagnostic information for a single error encountered by the device. The parameter code associated with error-event record indicates the relative time at which the error occurred. A higher parameter code indicates that the error event occurred later in time.

The content of the parameter value field of each log parameter is an ASCII character string which may describe the error event. The exact contents of the character string is not defined at this time.

8.10.9 Protocol Specific Port Log page (xxh)

The Protocol Specific Port log page (see Table 129) provides SCSI transport protocol specific parameters that are associated with the SCSI targets ports in the SCSI target device. This log page may be implemented in any logical unit, including the TARGET LOG PAGES well-known logical unit.

Table 129. Protocol Specific Port log page

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (18h) [1]							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3) [1]						
3								(LSB)

Protocol specific port log parameters [2]

4	First protocol specific port log parameter
:	:
n	Last protocol specific port log parameter

[1] The PAGE CODE and PAGE LENGTH fields are defined in Notes [1] and [2] of Table 112.

[2] The PROTOCOL SPECIFIC PORT LOG provides a list of Protocol Specific Port information, the format of which is given in Table 130.

Table 130. Protocol specific port log parameter format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PARAMETER CODE [a]						
1								(LSB)
2	DU [b]	DS [b]	TSD [b]	ETC [b]	TMC [b]		LBIN [b]	LP [b]
3	PARAMETER LENGTH (x-3) [c]							
	Reserved				PROTOCOL IDENTIFIER [d]			
4	SCSI transport protocol specific [e]							
255								

[a] The PARAMETER CODE field contains the relative target port identifier (see 8.10.9.1) of the target port for which the parameter data applies. Protocol specific log parameters for relative target ports numbered greater than 65 535 are not supported.

[b] The contents of the DU, DS, TSD, ETC, LBIN, and LP bits and the TMC field are defined in Table 128.

[c] The PARAMETER LENGTH field indicates the number of bytes remaining in the log parameter.

[d] The PROTOCOL IDENTIFIER field contain one of the values shown in Table 94 to indicate the SCSI transport protocol that defines the SCSI transport protocol specific data in this log parameter.

[e] The SCSI transport protocol specific data is defined by the corresponding SCSI transport protocol standard.

8.10.9.1 Relative target port identifier format

If the identifier type is relative target port (4h) and the ASSOCIATION field contains 1h, the four byte fixed length IDENTIFIER field shall have the format shown in Table 131. The CODE SET field shall be set to 1h and the IDENTIFIER LENGTH field shall be set to 4h. If the ASSOCIATION field does not contain 1h, use of this identifier type is reserved.

Table 131. Relative target port IDENTIFIER field format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	RELATIVE TARGET PORT						
3								(LSB)

The relative target port field (see Table 132) identifies the target port relative to other target ports in the device.

Table 132. Relative target port identifier values

Value	Description
0h	Reserved
1h	Relative port 1, historically known as port A
2h	Relative port 2, historically known as port B
3h - 7FFF FFFFH	Relative port 3 through 2 147 483 647
8000 0000H - FFFF FFFFH	Reserved

8.10.10 Non-Medium Error page (06h)

Log page code 06h specifies non-medium errors. The page format is shown in Table 133.

Table 133. Non-Medium Error page (06h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	PARAMETER CODE [1]							

[1] PARAMETER CODE 00h is the only code supported for this page and it represents the number of recoverable error events other than WRITE, READ, or VERIFY errors.

8.10.11 Self-Test Results Log page (10h)

The Self-Test Results Log page (see Table 134) provides the results from the twenty most recent self-tests (see Section 9.2). Results from the most recent self-test or the self-test currently in progress shall be reported in the first self-test log parameter; results from the second most recent self-test shall be reported in the second self-test log parameter; etc. If fewer than twenty self-tests have occurred, the unused self-test log parameter entries shall be zero filled.

Table 134. Self-Test Results Log Page format (10h)

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (10h) [1]							
1	Reserved							
2	(MSB)			PAGE LENGTH (0190h) [1]				
3								(LSB)

Self-test results log parameters

Byte	7	6	5	4	3	2	1	0
4 : 23	FIRST SELF-TEST RESULTS LOG PARAMETER [2] (most recent)							
:	:							
384 : 403	TWENTIETH SELF-TEST RESULTS LOG PARAMETER [2] (least recent)							

[1] The PAGE CODE and PAGE LENGTH fields are described in notes [1] and [2] of Table 112.

[2] Table 135 shows the format of one SELF-TEST LOG PARAMETER.

Table 135: Self-Test Results Log Parameter format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PARAMETER CODE (0001h to 0014h) [a]						
1								(LSB)
2	DU [b]	DS [b]	TSD [b]	ETC [b]	TMC [b]		LBIN [b]	LP [b]
3	PARAMETER LENGTH (10h) [c]							
4	SELF_TEST CODE [d]			Reserved	SELF-TEST RESULTS [e]			
5	SELF-TEST SEGMENT NUMBER [f]							
6	(MSB)	TIMESTAMP [g]						
7								(LSB)
8	(MSB)	ADDRESS OF FIRST FAILURE [h]						
15								(LSB)
16	Reserved				SENSE KEY [i]			
17	ADDITIONAL SENSE CODE [i]							
18	ADDITIONAL SENSE CODE QUALIFIER [i]							
19	vendor-specific							

[a] The PARAMETER CODE field identifies the log parameter being transferred. The PARAMETER CODE field for the results of the most recent self-test shall contain 0001h; the PARAMETER CODE field for the results of the second most recent test shall contain 0002h; etc.

[b] The values of the LOG PARAMETER control bits [2] through [8] for self-test results log parameters is specified in Table 136.

Table 136: Parameter control bits for self-test results log parameters

Bit	Value	Description
DU	0	Value provided by target
DS	0	target supports saving of parameter
TSD	0	target manages saving of parameter
ETC	0	No threshold comparison is made on this value
TMC	xxx	Ignored when ETC is 0
LBIN	1	The parameter is in binary format
LP	1	The parameter is a list parameter

[c] The PARAMETER LENGTH field shall contain 10h.

[d] The SELF-TEST CODE field contains the value in the Self-Test Code field of the SEND DIAGNOSTICs command that initiated this self-test (see SEND DIAGNOSTICs command).

[e] SELF-TEST RESULTS field values is shown in Table 137.

Table 137: Self-test results values

Value	Description
0h	Self-test completed without error
1h	The background self-test was aborted by the initiator using a SEND DIAGNOSTICs command with the Self-Test Code field set to 100b (Abort background self-test).
2h	The self-test routine was aborted by an initiator using a method other than a SEND DIAGNOSTICs command with the Self-Test Code field set to 100b (e.g., by a task management function, by a reset, or by issuing an exception command as defined in Section 9.2).
3h	An unknown error occurred while the target was executing the self-test and the target was unable to complete the self-test.
4h	The self-test completed with a failure in a test segment, and the test segment that failed is not known.
5h	The first segment of the self-test failed.
6h	The second segment of the self-test failed.
7h	Another segment of the self-test failed (see note [12] The Self-Test Segment Number field).
8h - Eh	Reserved.
Fh	The self-test is in progress.

[f] The SELF-TEST SEGMENT NUMBER field identifies the number of the segment that failed during the self-test. When the segment that failed cannot or need not be identified, this field shall contain 00h.

[g] The TIMESTAMP field contains the total accumulated power-on hours for the target at the time the self-test was completed. If the test is still in progress, the content of the Timestamp field shall be zero. If the power-on hours for the target at the time the self-test was completed is greater than FFFFh then the content of the Timestamp field shall be FFFFh.

[h] The ADDRESS OF FIRST FAILURE field contains information that locates the failure on the media. If the logical unit implements logical blocks, the content of the ADDRESS OF FIRST FAILURE field is the first LOGICAL BLOCK ADDRESS where a self-test error occurred. This implies nothing about the quality of any other logical block on the logical unit, since the testing during which the error occurred may not have been performed in a sequential manner. This value shall not change (e.g., as the result of block reassignment). The content of the ADDRESS OF FIRST FAILURE field shall be FFFFFFFFFFFFFFFFh if no errors occurred during the self-test or if the error that occurred is not related to an identifiable media address.

[i] The SENSE KEY, ADDITIONAL SENSE CODE, and ADDITIONAL SENSE CODE QUALIFIER fields may contain a hierarchy of additional information relating to error or exception conditions that occurred during the self-test represented in the same format used by the sense data (see REQUEST SENSE command).

8.10.12 Start-Stop Cycle Counter page (0Eh)

This section defines the optional Start-Stop Cycle Counter page (page code 0Eh). A device that implements the start-stop cycle counter page shall implement one or more of the defined parameters. Table 138 shows the start-stop cycle counter page with all parameters present.

Table 138. Start-Stop Cycle Counter page (0Eh)

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (0Eh) [1]							
1	Reserved							
2	(MSB)	PAGE LENGTH (24h) [1]						
3								(LSB)
4	(MSB)	PARAMETER CODE 0001h [2] DATE OF MANUFACTURE						
5								(LSB)
6	DU [3]	DS [3]	TSD [3]	ETC [3]	TMC [3]		LBIN [3]	LP [3]
7	PARAMETER LENGTH (06h)							
8	(MSB)	YEAR OF MANUFACTURE (4 ASCII characters) [2]						
11								(LSB)
12	(MSB)	WEEK OF MANUFACTURE (2 ASCII characters) [2]						
13								(LSB)
14	(MSB)	PARAMETER CODE 0002h [4] ACCOUNTING DATE						
15								(LSB)
16	DU [5]	DS [5]	TSD [5]	ETC [5]	TMC [5]		LBIN [5]	LP [5]
17	PARAMETER LENGTH (06h)							
18	(MSB)	ACCOUNTING DATE YEAR (4 ASCII characters) [4]						
21								(LSB)
22	(MSB)	ACCOUNTING DATE WEEK (2 ASCII characters) [4]						
23								(LSB)
24	(MSB)	PARAMETER CODE 0003h [6] SPECIFIED CYCLE COUNT OVER DEVICE LIFETIME						
25								(LSB)
26	DU [7]	DS [7]	TSD [7]	ETC [7]	TMC [7]		LBIN [7]	LP [7]
27	PARAMTER LENGTH (04h)							

Table 138. Start-Stop Cycle Counter page (0Eh)

Bit Byte	7	6	5	4	3	2	1	0
28	(MSB)	SPECIFIED CYCLE COUNT OVER DEVICE LIFETIME [6] (4-byte binary number)						
31								(LSB)
32	(MSB)	PARAMETER CODE 0004h [8] ACCUMULATED START-STOP CYCLES						
33								(LSB)
34	DU [7]	DS [7]	TSD [7]	ETC [7]	TMC [7]		LBIN [7]	LP [7]
35	PARAMETER LENGTH (04h)							
36 : 39	(MSB)	ACCUMULATED START-STOP CYCLES [8] (4 byte binary number						(LSB)

[1] The PAGE CODE and PAGE LENGTH fields are described in notes [1] and [2] of Table 112.

[2] The year and week in the year that the device was manufactured shall be set in the parameter field defined by parameter code 0001h. The date of manufacture shall not be saveable by the initiator using the LOG SELECT command. The date is expressed in numeric ASCII characters (30h–39h) in the form YYYYWW.

[3] The state of the PARAMETER CONTROL bits for parameter 0001h is specified in Table 139.

Table 139: Parameter control bits for date of manufacture parameters (0001h)

Bit	Value	Description
DU	0	Value provided by target
DS	1	target does not support saving of parameter
TSD	0	target manages saving of parameter
ETC	0	No threshold comparison is made on this value
TMC	xx	ignored when ETC is 0
LBIN	0	The parameter is in ASCII format
LP	1	The parameter is a list parameter

[4] The ACCOUNTING DATE specified by parameter code 0002h is a parameter that may optionally be savable using a LOG SELECT command to indicate when the device was placed in service. If the parameter is not yet set or is not setable, the default value placed in the parameter field shall be 6 ASCII blank characters (20h). The field shall not be checked for validity by the target.

[5] The state of the parameter control bits for parameter 0002h is specified in Table 140.

Table 140: Parameter control bits for date of manufacture parameters (0002h)

Bit	Value	Description
DU	0	Value provided by target
DS	0 or 1	target does not support saving of parameter
TSD	0	target manages saving of parameter
ETC	0	No threshold comparison is made on this value
TMC	xx	ignored when ETC is 0
LBIN	0	The parameter is in ASCII format
LP	1	The parameter is a list parameter

[6] The SPECIFIED CYCLE COUNT OVER DEVICE LIFETIME (parameter code 0003h) is a parameter provided by the target. The specified cycle count over device lifetime parameter shall not be saveable by the initiator using the LOG SELECT command. The parameter value is a 4-byte binary number. The value indicates how many stop-start cycles may typically be executed over the lifetime of the device without degrading the device’s operation or reliability outside the limits specified by the manufacturer of the device.

[7] The state of the PARAMETER CONTROL bits for parameter 0003h and 0004h is specified in Table 141.

Table 141: Parameter control bits for date of manufacture parameters (0003h and 0004h)

Bit	Value	Description
DU	0	Value provided by target
DS	1	target does not support saving of parameter
TSD	0	target manages saving of parameter
ETC	0	No threshold comparison is made on this value
TMC	xx	ignored when ETC is 0
LBIN	1	The parameter is in ASCII format
LP	1	The parameter is a list parameter

[8] The ACCUMULATED START-STOP CYCLES (parameter code 0004h) is a parameter provided by the target. The ACCUMULATED START-STOP CYCLES parameter shall not be saveable by the initiator using the LOG SELECT command. The parameter value is a 4-byte binary number. The value indicates how many start-stop cycles the device has detected since its date of manufacture. The time at which the count is incremented during a start-stop cycle is vendor-specific. For rotating magnetic storage devices, a single start-stop cycle is defined as an operational cycle that begins with the disk spindle at rest, continues while the disk accelerates to its normal operational rotational rate, continues during the entire period the disk is rotating, continues as the disk decelerates toward a resting state, and ends when the disk is no longer rotating. For devices without a spindle or with multiple spindles, the definition of a single start-stop cycle is vendor-specific. The count is incremented by one for each complete start-stop cycle. No comparison with the value of parameter 0003h shall be performed by the target.

8.10.13 Supported Log pages (00h)

The Supported Log pages (see Table 130) returns the list of log pages supported by the device. Targets that implement the LOG SENSE command shall implement this log page.

This page is not defined for the LOG SELECT command. This log page returns the list of supported log pages for the specified logical unit. Table 115 lists those pages that could be supported by the device.

Table 142. Supported Log pages

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (00h)							
1	Reserved							
2	(MSB)			PAGE LENGTH (n – 3) [1]				
3								(LSB)
4				SUPPORTED PAGE LIST [2]				
n								

[1] The PAGE LENGTH field specifies the length in bytes of the following SUPPORTED PAGE LIST.

[2] The SUPPORTED PAGE LIST field shall contain a list of all log page codes implemented by the target in ascending order beginning with page code 00h.

8.10.14 Temperature page (0Dh)

This clause defines the optional temperature log page (page code 0Dh). A device that implements the temperature page shall implement parameter 0000h. Parameter 0001h is optional and may be either omitted or set to a value indicating that the parameter is not defined. Table 143 shows the temperature page with all parameters present.

Table 143. Temperature page

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (0Dh)							
1	Reserved							
2	(MSB)			PAGE LENGTH (0Ch)				
3								(LSB)
4	(MSB)			PARAMETER CODE 0000h TEMPERATURE [1]				
5								(LSB)
6	DU	DS	TSD	ETC	TMC		LBIN	LP
7	PARAMETER LENGTH (02h)							
8	Reserved							
9	TEMPERATURE (degrees Celsius)							
10	(MSB)			PARAMETER CODE 0001h REFERENCE TEMPERATURE [2]				
11								(LSB)
12	DU	DS	TSD	ETC	TMC		LBIN	LP
13	PARAMETER LENGTH (02h)							
14	Reserved							
15	REFERENCE TEMPERATURE (degrees Celsius)							

[1] The temperature sensed in the device at the time the LOG SENSE command is performed shall be returned in the parameter field defined by parameter code 0000h. The one byte binary value specifies the temperature of the device in degrees Celsius. Temperatures equal to or less than zero degrees Celsius shall be indicated by a value of zero. If the target is unable to detect a valid temperature because of a sensor failure or other condition, the value returned shall be FFh. The temperature should be reported with an accuracy of plus or minus three Celsius degrees while the device is operating at a steady state within the environmental limits specified for the device. No comparison is performed between the temperature value specified in parameter 0000h and the reference temperature specified in parameter 0001h. The state of the parameter control bits for parameter 0000h is specified in Table 144.

Table 144: Parameter control bits for temperature parameters (0000h and 0001h)

Bit	Value	Description
DU	0	Value provided by target
DS	1	target does not support saving of parameter
TSD	0	target manages saving of parameter
ETC	0	No threshold comparison is made on this value
TMC	xx	Ignored when ETC is 0
LBIN	1	The parameter is in ASCII format
LP	1	The parameter is a list parameter

[2] A reference temperature for the device may optionally be provided by the device using parameter code 0001h. If no reference temperature is provided, the parameter may not be provided in the log page or alternatively, the reference temperature value may be set to the value of FFh. The one byte binary value should reflect the maximum reported sensor temperature in degrees Celsius at which the device will operate continuously without degrading the device's operation or reliability outside the limits specified by the manufacturer of the device. The reference temperature may change for vendor-specific reasons. The state of the parameter control bits for parameter 0001h is specified in Table 144 also.

8.11 MODE SELECT (6) command (15h)

The MODE SELECT (6) command (Table 145) provides a means for the initiator to specify medium, logical unit, or peripheral device parameters to the drive. The drive also implements the MODE SENSE (6) command (see Section 8.13.1). Initiators should issue MODE SENSE prior to MODE SELECT to determine supported pages, page lengths, and other parameters.

The drive maintains a common set of mode parameters shared by all initiators. If a SCSI initiator port sends a MODE SELECT command that changes any parameters that apply to other initiators, the drive generates a UNIT ATTENTION condition for all initiators except the one that issued the MODE SELECT command. The drive sets the additional sense code to Mode Parameters Changed.

Table 145. MODE SELECT (6) command (15h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	1	0	1	0	1
1	RESERVED [1]			PF [2]	Reserved			SP [3]
2	Reserved							
3	Reserved							
4	PARAMETER LIST LENGTH [4]							
5	CONTROL [5]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] The Page Format (PF) bit, when set to one, indicates the data sent by the initiator after the MODE SELECT Header and Block Descriptors (if any) complies with the Page Format. The PF bit, when set to zero, indicates the data sent after the MODE SELECT Header and the Block Descriptors (if any) are vendor unique. The drive does not interpret the PF bit. It assumes the Page Format mode.

[3] The Save Parameters (SP) bit, when set to one, requests that the drive save the savable pages. The format related parameters in the block descriptor, pages 3 and 4 are saved during a FORMAT command as well as a MODE SELECT command with SP = 1. The drive must update the Current mode values with parameters included with this command, save the Current values of the savable parameters, and report Good status only after the save operation is completed. The Saved parameters are not changed if an error is detected during the MODE SELECT command. When the SP bit is set to zero, the Saved parameter values are not changed.

[4] The PARAMETER LIST LENGTH specifies the length in bytes of the MODE SELECT parameter list that shall be transferred during the DATA OUT phase. A PARAMETER LIST LENGTH of zero indicates that no data shall be transferred. This condition shall not be considered as an error.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

8.11.1 MODE SELECT (6) parameter list

The MODE SELECT (6) parameter list (Table 146) contains a four-byte header, followed by zero or one block descriptor, followed by the pages of MODE SELECT parameters.

Acceptable values for the MODE SELECT parameter list for the drive are shown in Table 146.

The target terminates all the MODE SELECT commands with CHECK CONDITION status, sets the sense key to ILLEGAL REQUEST and sets the additional sense code to INVALID FIELD IN PARAMETER LIST, and does not change any mode parameters for the following conditions:

a. If the Strict mode is enabled (see note [8] of Table 155) and the initiator attempts to change any field that is not changeable by the host as reported by the target. In this case, no parameters are changed by this command. The target compares the parameters against the values as they were prior to this MODE SELECT command. (The host shall not be penalized by values not changeable by the host which have a SCSI target port “ripple change” as a result of this MODE SELECT.)

b. If the initiator attempts to send an unsupported value or a nonzero value to a reserved field in the MODE SELECT header, block descriptor, or any page header.

c. If a SCSI initiator port attempts to send a page with a length not equal to the parameter length reported for that page by the MODE SENSE command.

d. If the initiator attempts to send a value for a changeable parameter that is outside the range supported by the target and rounding is not implemented for that parameter. (When the ROUND bit equals one, the drive treats and reports rounded parameters as described in Section 7.8. When the ROUND bit equals zero, the drive shall round the parameter and handle command completion reporting as if the parameter had not been rounded.)

e. If the initiator sends a page descriptor with an unsupported page code value and the Strict mode is enabled. (When the STRICT bit is one, the drive checks for initiator attempts to change unchangeable parameters. If the drive detects an attempt, it rejects the command in the standard way, i.e., CHECK CONDITION status from drive, REQUEST SENSE from the initiator, and Illegal Request Sense key (5h) back from the drive. When the STRICT bit is zero, the drive ignores the values of the unchangeable parameters in a MODE SELECT command. The drive does not reject the command trying to change unchangeable parameters).

If the initiator sends a value for a changeable parameter that is outside the range supported by the target and rounding is implemented for that parameter, the target shall either:

a. round the parameter to an acceptable value and, if the ROUND bit is one, terminate the command as described in Section 7.8, or

b. round the parameter to an acceptable value and, if the ROUND bit equals zero, terminate the command as if an acceptable value had been sent from the initiator.

A SCSI target port may alter any mode parameter in any mode page (even parameters reported as non-changeable) as a result of changes to other mode parameters.

Table 146. MODE SELECT (6) parameter list

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
	Reserved							
1	0	0	0	0	0	0	0	0
	MEDIUM TYPE [1]							
2	0	0	0	0	0	0	0	0
	Reserved							
3	BLOCK DESCRIPTOR LENGTH (either 0 or 8 decimal) [2]							

Block Descriptor

0	DENSITY CODE or NUMBER OF BLOCKS (MSB) [3] [4]							
1	NUMBER OF BLOCKS (MSB) [3] [4]							
2	NUMBER OF BLOCKS [4]							
3	NUMBER OF BLOCKS (LSB) [4]							
4	0	0	0	0	0	0	0	0
5	BLOCK LENGTH (MSB) [5]							
6	BLOCK LENGTH [5]							
7	BLOCK LENGTH (LSB) [5]							

Parameter Information [6]

0-n	MODE SELECT PAGE HEADERS AND THEIR PARAMETERS (Tables 147 and 148)

[1] The MEDIUM TYPE field shall be 00h to define the default type direct access device.

[2] The BLOCK DESCRIPTOR LENGTH specifies the length in bytes of the Block Descriptor. It is equal to the number of bytes in the Block Descriptor (either 0 or 8) and does not include the page headers and mode parameters. A BLOCK DESCRIPTOR LENGTH of zero indicates that no block descriptors shall be included in the parameter list. This condition shall not be considered an error.

[3] a. For drives that do not support capacity programming, the DENSITY CODE shall be 00h to define the default density of medium.

b. If the drive supports capacity programming (see note [4]), byte 0 may be useable as MSB part of the number in bytes 1, 2, and 3, if the drive's capacity capability requires it. For drives whose programmed capacity or maximum designed capacity does not require the use of byte 0, byte 0 will always be 00h. (See also note [4b]).

[4] a. For drives that do not support capacity programming, these bytes are always zero.

b. A capacity value greater than the product or customer defined maximum but less than the drive's physical maximum will result in the drive taking the specified capacity. Any request that is beyond the physical maximum will result in an ILLEGAL REQUEST. The product-defined maximum is the drive's default capacity and can be determined by setting bytes 0, 1, 2, and 3 to 0xFFFFFFFF.

[5] BLOCK LENGTH specifies the length in number of bytes for each logical block described by the Block Descriptor. Set to desired sector size before a Format. (Valid values are even numbered sizes from 180 to 4096. Not all drives can format down to 180; some have a minimum of 256).

[6] See MODE SENSE (6) command (Section 8.13) for detailed descriptions of the Mode Select/Sense pages.

8.11.2 MODE SELECT (6) page descriptors

The rest of the MODE SELECT parameters are organized into pages that group the parameters by function. The parameter definitions are the same as those described in the MODE SENSE (6) command (Section 8.13) and are not repeated here.

Table 147. MODE SELECT (6) page descriptor header

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	PAGE CODE					
1	PAGE LENGTH							
2-n	MODE PARAMETERS							

Each page of mode parameters begins with a two byte page descriptor header. The PAGE CODE identifies the page of mode parameters that is being transferred. The PAGE LENGTH indicates the number of additional bytes of mode parameters contained in this page. The number of additional bytes sent must always match the PAGE LENGTH value.

The drive only verifies MODE SELECT Data that is defined as changeable by the drive. The various drives support the following Page Codes.

Note. See individual drive's Product Manual, Volume 1, section "SCSI Interface commands supported," for a table showing the mode pages that a particular drive implements. The table shows the default parameters for pages that are implemented and shows which mode parameters are changeable by that drive model.

Table 148. MODE SENSE page codes supported

Page Code	Description
00h	Unit Attention page parameters (should be sent last in a group of Mode pages)
01h	Error Recovery parameters
02h	Disconnect/Reconnect Control parameters
03h	Format parameters
04h	Rigid Drive Geometry parameters
07h	Verify Error Recovery parameters
08h	Caching parameters
0Ah	Control Mode page
0Ch	Notch page
1Ah	Power Condition page
10h	Xor Control Mode page
19h	SCSI Port Control Mode page
1Ch	Informational Exceptions Control page
3Fh	Return all supported pages

The detailed information can be obtained by issuing the MODE SENSE command requesting changeable values.

Note. There may be implicit associations between parameters defined in the pages and block descriptors. The block length affects the optimum values (the values that achieve best performance) for the sectors per track, bytes per physical sector, track skew factor, and cylinder skew factor fields in the format parameters page. In this case, the drive may change parameters not explicitly sent with the MODE SELECT command. A subsequent MODE SENSE command would provide information on these changes.

8.12 MODE SELECT (10) command (55h)

The MODE SELECT (10) command (Table 149) provides a means for the initiator to send a list of drive operating mode parameters to the drive. Initiators should issue MODE SENSE prior to MODE SELECT to determine supported pages, page lengths, and other parameters.

Table 149. MODE SELECT (10) command (55h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	0	1	0	1	0	1
1	RESERVED [1]			PF [2]	Reserved			SP [3]
2 : 6	Reserved							
7	(MSB)			PARAMETER LIST LENGTH [4]				
8								(LSB)
9	CONTROL [5]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] The Page Format (PF) bit, when set to one, indicates the data sent by the initiator after the MODE SELECT Header and Block Descriptors (if any) complies with the Page Format. The PF bit, when set to zero, indicates the data sent after the MODE SELECT Header and the Block Descriptors (if any) are vendor unique. The drive does not interpret the PF bit. It assumes the Page Format mode.

[3] The Save Parameters (SP) bit, when set to one, requests that the drive save the savable pages. The format related parameters in the block descriptor, pages 3 and 4 are saved during a FORMAT command as well as a MODE SELECT command with SP = 1. The drive must update the current mode values with parameters included with this command, save the current values of the savable parameters, and report GOOD status only after the save operation is completed. The saved parameters are not changed if an error is detected during the MODE SELECT command. When the SP bit is set to zero, the saved parameter values are not changed.

[4] The PARAMETER LIST LENGTH specifies the length in bytes of the MODE SELECT parameter list that shall be transferred during the DATA OUT phase. A PARAMETER LIST LENGTH of zero indicates that no data shall be transferred. This condition shall not be considered as an error.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

8.12.1 MODE SELECT (10) parameter list

The MODE SELECT (10) parameter list (Table 150) contains an eight-byte header, followed by zero or one block descriptor, followed by the pages of MODE SELECT parameters.

Acceptable values for the MODE SELECT parameter list for the drive are shown in Table 150.

The target terminates all the MODE SELECT commands with CHECK CONDITION status, sets the sense key to ILLEGAL REQUEST and sets the additional sense code to INVALID FIELD IN PARAMETER LIST, and does not change any mode parameters for the following conditions:

a. If the Strict mode is enabled (see note [8] of Table 155) and the initiator attempts to change any field that is not changeable by the host as reported by the target. In this case, no parameters are changed by this command. The target compares the parameters against the values as they were prior to this MODE SELECT command. (The host shall not be penalized by values not changeable by the host which have a SCSI target port “ripple change” as a result of this MODE SELECT.)

b. If the initiator attempts to send an unsupported value or a nonzero value to a reserved field in the MODE SELECT header, block descriptor, or any page header.

c. If a SCSI initiator port attempts to send a page with a length not equal to the parameter length reported for that page by the MODE SENSE command.

d. If the initiator attempts to send a value for a changeable parameter that is outside the range supported by the target and rounding is not implemented for that parameter (see note [7] of Table 155).

e. If the initiator sends a page descriptor with an unsupported page code value and the Strict mode is enabled (see note [8] of Table 155).

If the initiator sends a value for a changeable parameter that is outside the range supported by the target and rounding is implemented for that parameter, the target shall either:

a. round the parameter to an acceptable value and, if Round is one, terminate the command as described in Section 7.8, or

b. round the parameter to an acceptable value and, if Round equals zero, terminate the command as if an acceptable value had been sent from the initiator.

A SCSI target port may alter any mode parameter in any mode page (even parameters reported as non-changeable) as a result of changes to other mode parameters.

Table 150. MODE SELECT (10) parameter list

Bit Byte	7	6	5	4	3	2	1	0
0,1	0	0	0	0	0	0	0	0
	Reserved							
2	0	0	0	0	0	0	0	0
	MEDIUM TYPE [1]							
3,4,5	0	0	0	0	0	0	0	0
	Reserved							
6,7	BLOCK DESCRIPTOR LENGTH (after 0 or 8 decimal) [2]							

Block Descriptor Data

0	0	0	0	0	0	0	0	0
	DENSITY CODE [3]							
1	0	0	0	0	0	0	0	0
	NUMBER OF BLOCKS (MSB) [3] [4]							
2	0	0	0	0	0	0	0	0
	NUMBER OF BLOCKS [4]							

Bit Byte	7	6	5	4	3	2	1	0
3	0	0	0	0	0	0	0	0
	NUMBER OF BLOCKS (LSB) [4]							
4	0	0	0	0	0	0	0	0
5	BLOCK LENGTH (MSB) [5]							
6	BLOCK LENGTH [5]							
7	BLOCK LENGTH (LSB) [5]							

Parameter Information [6]

0-n	MODE SELECT PAGE HEADERS AND THEIR PARAMETERS (Tables 147 and 148)

[1] The MEDIUM TYPE field shall be 00h to define the default type direct access device.

[2] The BLOCK DESCRIPTOR LENGTH specifies the length in bytes of the Block Descriptor. It is equal to the number of bytes in the Block Descriptor (either 0 or 8) and does not include the page headers and mode parameters. A BLOCK DESCRIPTOR LENGTH of zero indicates that no block descriptors shall be included in the parameter list. This condition shall not be considered an error.

[3] a. For drives that do not support capacity programming, the DENSITY CODE shall be 00h to define the default density of medium.

b. If the drive supports capacity programming (see note [4]), byte 0 is useable as MSB part of the number in bytes 1, 2, and 3. For drives whose capacity does not require the use of byte 0, byte 0 will always be 00h.

[4] a. For drives that do not support capacity programming, these bytes are always zero.

b. A value of zero in bytes 1, 2, and 3 indicates that the drive shall not change the capacity it is currently formatted to have. Any other value in these bytes is ignored by drives not having the capacity programming feature. For drives that have the capacity programming capability (see individual drive’s Product Manual, Volume 1), a number in bytes 0, 1, 2, and 3 that is less than the maximum number of LBAs changes the drive capacity to the value in the block descriptor bytes 0, 1, 2, and 3. A value greater than the maximum number of LBAs is rounded down to the maximum capacity.

[5] BLOCK LENGTH specifies the length in number of bytes for each logical block described by the Block Descriptor. Set to desired sector size before a Format. (Valid values are even numbered sizes from 180 to 4096. Not all drives can format down to 180; some have a minimum of 256).

[6] See MODE SENSE (10) command (Section 8.14) for detailed descriptions of the Mode Select/Sense pages.

8.13 MODE SENSE (6) command (1Ah)

The MODE SENSE (6) command provides a means for the drive to report its medium, logical unit, or peripheral device parameters to the initiator. It is a command complementary to the MODE SELECT (6) command.

Table 151. MODE SENSE (6) command (1Ah)

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (1Ah)							
1	0	0	0	0	0	0	0	0
	RESERVED [1]				DBD [2]	Reserved		
2	PC [3]		PAGE CODE [4]					
3	SUBPAGE CODE [4]							
4	ALLOCATION LENGTH [5]							
5	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Block Descriptors (DBD) bit of zero indicates that the drive may return zero or more block descriptors in the returned MODE SENSE data, at the drive's discretion. Seagate SCSI-2 products return one block descriptor if the DBD bit is zero. A DBD bit of one specifies that the drive shall not return any block descriptors in the returned MODE SENSE data.

[3] The content of mode parameter bytes is determined by the value of the PC (Page Control) bits specified in CDB byte 2, bits 6 and 7. The drive shall return the same PAGE LENGTH for each supported page regardless of the value of PC. The PC field is defined in Table 152.

Table 152. Page Control field bits' affect on mode parameters returned

Bit 7	Bit 6	
0	0	Return current values. The current values are the values currently being used by the drive to control its operation. After a POWER ON RESET, a HARD RESET, or a BUS DEVICE RESET message the current values are equal to the Saved values (if Saved values can be retrieved) or the Default values (if Saved values cannot be retrieved). The current value of a parameter is updated by a MODE SELECT command if the MODE SELECT Command ends with GOOD status returned.
0	1	Return changeable values. The changeable values of any page is a mask that indicates the parameters that shall be changed via a MODE SELECT command and the parameters that shall not. Each returned parameter byte shall contain ones where a field or bit may be changed and zeros where a field or bit may not be changed.
1	0	Return default values. The Default values are the values to which the drive sets the Current values after a reset condition unless valid Saved values are available.
1	1	Return saved values. The saved values are the values the drive stores in nonvolatile memory. The Saved values of any changeable parameter can be set to new values via a MODE SELECT command with the SMP bit set to one. For non-changeable parameters, the Default value is used.

The Block descriptor contains its normal values regardless of the value of the PC field. Unsupported fields or bits within a page are returned as zeros for all PC field values.

[4] The PAGE CODE and SUBPAGE CODE fields specify which mode pages and subpages to return. Refer to Section 8.13.14 on page 271 (Port Control Mode page) for the subpage code used by the Port Control Mode page.

Page Codes that may be supported by the drive are summarized here (see individual drive's Product Manual, Volume 1):

Page Code	Description
00h	Unit Attention page parameters (returned last of the pages)
01h	Error Recovery parameters
02h	Disconnect/Reconnect Control parameters
03h	Format parameters
04h	Rigid Drive Geometry parameters
07h	Verify Error Recovery page parameters
08h	Caching Parameters page
0Ah	Control Mode page
0Ch	Notch page
1Ah	Power Condition page
10h	Xor Control Mode page
19h	Port Control Mode page
1Ch	Informational Exceptions Control page
3Fh	Return all supported pages

[5] The ALLOCATION LENGTH specifies the number of bytes that the initiator has allocated for returned MODE SENSE data. An ALLOCATION LENGTH of zero indicates that no MODE SENSE data shall be transferred. This condition shall not be considered as an error. Any other value indicates the maximum number of bytes that shall be transferred. The drive shall terminate the DATA IN phase when ALLOCATION LENGTH bytes have been transferred or when all available MODE SENSE data has been transferred to the initiator, whichever is less.

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

Table 153. MODE SENSE (6) parameter list

Bit Byte	7	6	5	4	3	2	1	0
0	SENSE DATA LENGTH [1]							
1	0	0	0	0	0	0	0	0
	MEDIUM TYPE [2]							
2	WP [3]	0	0	DPO-FUA [8]	0	0	0	0
		Reserved			Reserved			
3	0	0	0	0	1	0	0	0
	BLOCK DESCRIPTOR LENGTH (8 decimal) [4]							

Block Descriptor Data

0	DENSITY CODE (MSB) [5] [6]							
1	NUMBER OF BLOCKS (MSB) [6]							
2	NUMBER OF BLOCKS [6]							
3	NUMBER OF BLOCKS (LSB) [6]							
4	0	0	0	0	0	0	0	0
	Reserved							
5	BLOCK LENGTH (MSB) [7]							
6	BLOCK LENGTH [7]							
7	BLOCK LENGTH (LSB) [7]							

Parameter Information

0-n	MODE SENSE PAGE HEADERS AND THEIR PARAMETERS

[1] The SENSE DATA LENGTH specifies the length in bytes of the following MODE SENSE data that is available to be transferred during the DATA IN phase. The SENSE DATA LENGTH does not include itself.

[2] The drive supports only 00h (default medium) in the MEDIUM TYPE field.

[3] A Write Protect (WP) bit of zero indicates the medium is write enabled. A WP bit of one indicates the medium is write protected.

[4] The BLOCK DESCRIPTOR LENGTH specifies the length in bytes of the Block Descriptor. It is equal to the number of bytes in the Block Descriptor (8) and does not include the page headers and mode parameters, if any. The drive sends one Block Descriptor.

Each Block Descriptor specifies the medium characteristics for all or part of a logical unit. Each Block Descriptor contains a DENSITY CODE, a NUMBER OF BLOCKS, and a BLOCK LENGTH.

[5] a. Drives that do not support capacity programming have only 00h (default density) in the DENSITY CODE field.

b. Drives that support capacity programming may have a value in this field that states either the logical or actual capacity of the drive.

[6] a. For drives that do not support capacity programming, the NUMBER OF BLOCKS field specifies the number of logical blocks of the medium that meets the DENSITY CODE and BLOCK LENGTH in the Block Descriptor. A NUMBER OF BLOCKS of zero indicates that all of the remaining logical blocks of the logical unit have the medium characteristics specified by the Block Descriptor.

b. For drives that have capacity programming capability, these bytes do not report back the drive capacity on some models. These bytes are always zero on those models. Some models do report drive capacity in bytes 0, 1, 2, and 3, so those bytes will be nonzero. See individual drive's Product Manual, Volume 1, for the drive of interest.

[7] The BLOCK LENGTH, as defined after a format function, specifies the length in bytes of each logical block described by the Block Descriptor. Default is 512 if no MODE SELECT command is received before the Format command. The usual valid values are 256 through 4096. Some drive products can format down to 180 bytes per sector. Some products can only format an even numbered value of bytes per sector (180-4096).

[8] When used with the MODE SENSE command, a DPO-FUA bit of zero indicates that the target does not contain a cache memory or does not support the DPO and FUA bits. A DPO-FUA bit of one indicates that the target supports the DPO and FUA bits.

8.13.1 MODE SENSE page descriptor header

Each page of mode parameters (for the MODE SENSE command) begins with a two-byte page descriptor header. The PAGE CODE identifies the page of mode parameters that is being transferred. The PAGE LENGTH indicates the number of additional bytes of mode parameters being sent by the drive. The parameter bit values are left blank herein, because they may be different for each drive model.

Note. See individual drive's Product Manual, Volume 1, for a table giving the MODE SENSE parameter values that are applicable to the drive model of interest. The tables in Volume 1 also show which parameters are changeable in the drive model of interest and which are not.

Multiple pages of mode parameters may be transferred in one MODE SENSE DATA IN phase (using PAGE CODE 3Fh). If a non-supported page code is requested by the Initiator, the drive terminates the command with CHECK CONDITION status, sets the sense key to 05, ILLEGAL REQUEST, and sets the additional sense code to 24, INVALID FIELD IN PARAMETER LIST.

The Parameters Savable (PS) bit, when set to one, indicates if the page contains savable parameters. When the PS bit is set to zero, none of the parameters within the page are savable. Since the parameters within pages 3 and 4 are always saved during format commands (but not via a MODE SELECT command with the SMP bit set to 1), these pages return a one for the PS bit.

Table 154. MODE SENSE page descriptor header

Bit Byte	7	6	5	4	3	2	1	0
0	PS	0	PAGE CODE [1]					
1	PAGE LENGTH [1]							
2–n	MODE PARAMETERS [1]							

[1]

Page Code	Page Description	Reference Table	Page
00h	Unit Attention Parameters	155	240
01h	Error Recovery	156	242
02h	Disconnect/Reconnect Control	157	245
03h	Format Parameters	158	248
04h	Rigid Drive Geometry	159	251
07h	Verify Error Recovery	160	254
08h	Caching Parameters	161	257
0Ah	Control Mode	162/164	261/262
0Ch	Notch	166	264
1Ah	Power Condition	167	266
10h	Xor Control Mode	168	267
19h	Port Control Mode	177	278
1Ch	Informational Exceptions Control	177	278

8.13.2 Unit Attention parameters page (00h)

The Unit Attention parameters page is the last page to be reported by the drive.

Table 155. Unit Attention parameters page (00h)

Bit Byte	7	6	5	4	3	2	1	0
0	1 PS [12]	0	0 PAGE CODE (00h)	0	0	0	0	0
1	PAGE LENGTH (in bytes) [1]							
2 Default	PM[2]	SSM [3]	INQUIRY LENGTH [4]	UNIT ATTENTION [5]	DFUA [6]	ROUND [7]	STRICT [8]	SCSI-2 [9]
Changeable	[11]							
3 Default	Reserved [10]	SELF SEEK [13]	Reserved [10]					
4	Reserved				JIT3	JIT2	JIT1	JIT0
					[14]			
5	Reserved [11]							
6	Reserved [11]							
7	Reserved [11]							

[1] The PAGE LENGTH field specifies the length in bytes of the mode parameters that follow. If the initiator does not set this value to the value that is returned for the page by the MODE SENSE command, the drive shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST with the additional sense code set to Invalid Field In Parameter List. The drive is permitted to implement a mode page that is less than the full page length defined by this specification, provided no field is truncated and the page length field correctly specifies the actual length implemented. If the STRICT bit equals zero and if the page length specified by the initiator is shorter than the actual page length, then the parameters are transferred and the command ends with GOOD status if no other items cause the command to be rejected.

Caution: Utilization of this forgiving option by a SCSI initiator port that does not analyze the impact of the truncation could adversely affect data integrity.

[2] The PM (Performance Mode) bit is used to control the drive’s cache management algorithm to allow best performance in different types of systems. It is the initiator’s responsibility to determine which setting is best for that system. A PM value of 1 indicates that the number of cache segments is fixed to the value set in mode page 8. A PM value of 0 indicates that the drive will optimize the number of segments depending on the command activity observed by the drive. The number of segments value (in made page 8) is ignored with the PM value is 0.

[3] If the enable Synchronous Select Mode (SSM) bit equals one, the drive initiates WDTR and SDTR messages when it recognizes that one may be required (after reset, reset message, or power cycle). If the

SSM bit equals zero, the drive does not initiate WDTR or SDTR regardless of negotiated conditions prior to reset, reset message, or power cycle.

[4] When the INQUIRY Length bit is set to 1, the standard INQUIRY data available to a host is limited to the 36 bytes required by the SCSI-2 specification. When the IL bit is reset (0), 148 bytes of standard INQUIRY data are available. The Additional Length field in byte 4 of the INQUIRY data is updated to reflect the actual number of additional bytes available.

[5] When the UNIT ATTENTION bit is set to 1, then UNIT ATTENTION is logged in sense only; no CHECK CONDITION status is presented following any reset. When this bit is Reset (0), then Check Condition is presented for all affected initiators following a reset until REQUEST SENSE is issued by each initiator (as per current operation).

[6] When the Disable Force Unit Access (DFUA) bit is set to 1, the drive ignores the FUA bit in READ and WRITE commands. This can result in better drive performance in some circumstances. When the DFUA is set to 0, the drive obeys the FUA setting in READ and WRITE commands.

[7] When the ROUND bit equals one, the drive treats and reports rounded parameters as described in Section 7.8. When ROUND equals zero, the drive shall round the parameter and handle command completion reporting as if the parameter had not been rounded.

[8] When the STRICT bit is a one, the drive checks for initiator attempts to change unchangeable parameters. If the drive detects an attempt, it rejects the command in the standard way, i.e., CHECK CONDITION status from drive, REQUEST SENSE from the initiator, and Illegal Request Sense key (5h) back from the drive. When the STRICT bit is zero, the drive ignores the values of the unchangeable parameters in a MODE SELECT command. The drive does not reject the command trying to change unchangeable parameters.

[9] When set to one, the SCSI-2 bit changes the following SCSI-3 features from their SCSI-3 definition to the SCSI-2 definition. When S2 equals zero, the following features remain as specified in other portions of this specification:

a. Control Mode Page (0Ah) Length from 0Ah to 06h.
b. Caching Page (08h) Length from 12h to 0Ah.

[10] These bits are reserved for future compatibility with Seagate host adapters. Though they presently may be changeable, (see note [9]) these bits do not control anything, unless the individual drive’s Product Manual, Volume 1, indicates that they do and defines their use in the MODE SENSE Data section.

[11] See individual drive’s Product Manual, Volume 1, MODE SENSE Data section for a table showing codes that indicate which of these bits are changeable by the host using the MODE SELECT command.

[12] A Parameter Savable (PS) bit of one indicates that the drive is capable of saving the page in a nonvolatile vendor-specific location (used only with MODE SENSE command).

[13] If the Self Seek bit is set to one, the drive will enter self seek mode for testing purposes. Such testing could include, but is not limited to, power dissipation and acoustics. While in this mode, the drive will accept SCSI commands and will process them in between the self seek operations, including a MODE SELECT to turn this bit back off. As such, this bit should be off for normal drive operations. If this bit is set to zero, the drive will not self seek; normal operating mode.

[14] The four JIT (Just In Time) bits allow you to enable and disable certain seek speeds. JIT0 represents the fastest seek type used by the drive, JIT1 represents the second fastest, JIT2 represents the third fastest, and JIT3 represents the slowest seek type. You can use these bits to reduce acoustics by disabling the fastest seeks. This can also reduce power consumption (from seek activity). These JIT settings only affect user read and write operations. Background drive operations and user seek commands will always use the fastest seek type. When the bit is set to 1, the drive is allowed to use this seek type in its seek speed algorithm. When the bit is set to 0, the drive is not allowed to use this seek type in its seek speed algorithm. If all JIT bits are set to zero, the drive enables JIT0 only. If all bits are set to one, the drive firmware selects the slowest (quietest) seek speed that does not hurt performance.

8.13.3 Error Recovery page (01h)

The drive Error Recovery page implementation is defined in Table 156. This table summarizes the function, the default value, and changeability status for each byte/bit.

Table 156. Error Recovery page (01h)

Bit Byte	7	6	5	4	3	2	1	0
Page Descriptor Header								
0	1	0	0	0	0	0	0	1
	PS [1]		PAGE CODE (01h)					
1	PAGE LENGTH (0Ah)							
Error Recovery Parameters								
2 Default	AWRE [3]	ARRE [4]	TB [5]	RC [6]	EER [7]	PER [8]	DTE [9]	DCR [10]
Changeable	[2]							
3 Default	READ RETRY COUNT [11]							
Changeable	[2]							
4 Default	CORRECTION SPAN (bits) [12]							
Changeable	[2]							
5 Default	HEAD OFFSET COUNT [13]							
Changeable	[2]							
6 Default	DATA STROBE OFFSET COUNT [14]							
Changeable	[2]							
7 Default	Reserved [17]							
Changeable	[2]							
8 Default	WRITE RETRY COUNT [15]							
Changeable	[2]							
9 Default	Reserved							

Bit Byte	7	6	5	4	3	2	1	0
Changeable	[2]							
10	(MSB)			RECOVERY TIME LIMIT [16]				
11								(LSB)
Changeable	[2]							

[1] The returned Parameter Savable (PS) bit of 1 indicates that page 01h parameter data is savable. This bit is not used with the MODE SELECT command.

[2] A value of zero means this bit function is not directly changeable by a SCSI initiator port, a value of 1 means the bit function is directly changeable by a SCSI initiator port (see MODE SELECT command). See individual drive’s Product Manual, Volume 1, section showing changeable values.

[3] The Automatic Write Reallocation of defective data blocks Enabled (AWRE) bit, when set to one, allows the drive to automatically relocate bad blocks detected during write operations. The drive performs the automatic write reallocation only if the drive has the valid data (e.g., original data in the buffer or recovered from the medium). The valid data is placed in the reallocated block. This function doesn’t apply to the FORMAT UNIT command. When set to zero, the drive shall not perform automatic reallocation but shall create CHECK CONDITION status with sense key of Medium Error instead.

[4] The Automatic READ Reallocation of defective data blocks Enabled (ARRE) bit, when set to one, allows the drive to automatically relocate bad blocks detected during READ operations. Automatic reallocation is performed only if the drive successfully recovers the data and is able to place it in the reallocated block. When set to zero, the drive shall not perform automatic reallocation but shall create CHECK CONDITION status with sense key of Medium Error instead.

[5] The Transfer Block (TB) bit, when set to one, indicates the data block that is not recovered shall be transferred to the initiator. When set to zero, the failing data block shall not be transferred.

[6] The READ Continuous (RC) bit, when set to one, requests the drive to transfer the requested data length without adding delays (for retries or ECC correction) that may be required to ensure data integrity. The drive may send erroneous data in order to maintain the continuous flow of data. This bit shall override the DTE bit if it is set. RC bit has priority also over ARRE, AWRE, EER, DCR, and PER bits. If the RC bit is set to one, no Auto Reallocation will be attempted. When set to zero, recovery actions during data transfer are allowed. This bit is set to zero and is not changeable in most if not all of the drive models covered by this manual. See individual drive’s Product Manual, Volume 1, section showing changeable values.

[7] The Enable Early Recovery (EER) bit, when set to one, allows the drive to apply maximum T level ECC correction on the fly, before attempting other retry mechanisms. SEEK error retries and message system errors are not affected by this bit. When this bit is set, the DCR bit must be zero. When the EER bit is set to zero, the drive shall apply ECC correction before other retry mechanisms, but shall perform only normal T level ECC corrections on the fly (Normal T level is product-specific, but is typically max T–1). In either case, any successful correction on the fly will not be reported to the host as an error.

[8] The Post Error (PER) bit, when set to one, indicates the drive reports CHECK CONDITION status and appropriate sense key for any recovered errors encountered. Reporting of unrecoverable errors has priority over reporting of recoverable errors. When set to zero, any errors recovered within the limits established by the other Error Recovery Flags are not reported. Any unrecoverable errors are reported.

[9] The Disable Transfer on Error (DTE) bit is valid only when the PER bit is set to one. When the DTE bit is set to one, it indicates the drive terminates data transfer even for recoverable errors (the drive will transfer the data for the recovered error before terminating the transfer). When DTE is set to zero, data transfer continues if recoverable errors are encountered. If the PER bit is one and the DTE bit is zero, recoverable errors are reported after all data has been transferred.

[10] The Disable Correction (DCR) bit, when set to one, indicates that only on-the-fly ECC correction shall be applied. The exact behavior is product specific. When set to zero, maximum ECC correction shall be applied if correction is possible.

[11] The READ RETRY COUNT sets up the maximum amount of error recovery effort to be applied for each LBA that could not be recovered during a Read operation. The hex value in this field specifies the maximum error recovery level that the drive applies during a Read operation to the recovery of an LBA needing recovery effort. Each level may consist of multiple error recovery steps. See individual drive's Product Manual, Volume 1, for more details on the levels of error recovery available.

[12] The CORRECTION SPAN is the size of the largest read data error, in bits, on which ECC correction is to be attempted. Errors longer than this span are reported as unrecoverable. This value is drive dependent. If this field is zero, the drive uses its default value.

[13] The drive HEAD OFFSET COUNT is a default of zero and not changeable to signify that this feature is not programmable by the initiator. Head offsets are performed as part of the drive's retry algorithms.

[14] The drive DATA STROBE OFFSET COUNT is a default of zero and not changeable to signify that this feature is not programmable by the initiator. Data strobe offsets are performed as part of the drive's retry algorithms.

[15] The WRITE RETRY COUNT sets up the maximum amount of error recovery to be applied for each LBA that could not be recovered during a Write operation. The hex value in this field specifies the maximum error recovery level that the drive applies during a Write operation to the recovery of an LBA needing recovery effort. Each level may consist of multiple error recovery steps. See individual drive's Product Manual, Volume 1, for more details on the levels of error recovery available.

[16] The RECOVERY TIME LIMIT field (bytes 10 and 11) specifies the maximum time in milliseconds that the host allows the drive to spend in error recovery efforts during the execution of a command. The READ and WRITE RETRY count can also be set to limit the amount of time the drive spends in error recovery of individual LBAs. The total of all times used to recover individual LBAs in the block called for by a command cannot exceed the RECOVERY TIME LIMIT value in bytes 10 and 11. Once the drive has reached the error recovery time limit for a particular command, the command ends with a CHECK CONDITION status and an unrecovered error is reported. A RECOVERY TIME LIMIT of FFFFh or 0000h means that the command recovery time is unlimited. A value of 0001h means that no time shall be spent in error recovery. A changeable RECOVERY TIME LIMIT is not supported on all drives supported by this manual. See individual drive's Product Manual, Volume 1, Mode page changeable bit settings for Mode page 01h, bytes 10 and 11.

[17] In SCSI-1 mode of operation, this byte is the recovery time limit value (see note [16]).

8.13.4 Disconnect/Reconnect Control page (02h)

The Disconnect/Reconnect mode page provides the applications client the means to tune the performance of the SCSI parallel interface. The Disconnect/Reconnect page implementation is defined in Table 157. This table summarizes the function and defines the default value and changeable status.

Table 157. Disconnect/Reconnect Control page (02h)

Bit Byte	7	6	5	4	3	2	1	0
Page Descriptor Header								
0	1	0	0	0	0	0	1	0
	PS [1]		PAGE CODE (02h)					
1	PAGE LENGTH (0Eh)							
Disconnect/Reconnect Control Parameters								
2 Default	BUFFER FULL RATIO [3] [4]							
Changeable	[2]							
3 Default	BUFFER EMPTY RATIO [3] [5]							
Changeable	[2]							
4 Default	BUS INACTIVITY LIMIT (MSB) [6]							
Changeable	[2]							
5 Default	BUS INACTIVITY LIMIT (LSB) [6]							
Changeable	[2]							
6,7 Default	DISCONNECT TIME LIMIT [7]							
Changeable	[2]							
8,9 Default	CONNECT TIME LIMIT [8]							
Changeable	[2]							
10,11 Default	(MSB)			MAXIMUM BURST SIZE [9]				
								(LSB)
Changeable	[2]							

Bit Byte	7	6	5	4	3	2	1	0
12 Default	EMDP [11]	FAIR ARBITRATION [13]			DIMM [12]	DTDC [10]		
Changeable	[2]							
13	Reserved							
14	(MSB)	FIRST BURST SIZE [14]						
15								(LSB)
Changeable	[2]							

[1] The PS (Parameter Savable) bit of 1 indicates that the page 02h parameter data is savable.

[2] A changeable value of zero means this function is not directly changeable by a SCSI initiator port. A value of 1 means the bit function is directly changeable by a SCSI initiator port (see MODE SELECT command). See individual drive's Product Manual, Volume 1, MODE SENSE data section for table showing changeable values.

[3] Both ratio parameters are the numerators of a fractional multiplier that has 256 (100h) as its denominator.

[4] The BUFFER FULL RATIO indicates, on READ commands, how full the drive's buffer shall be before attempting a reselection. This value is changeable by a SCSI initiator port, but the value is ignored by the drive. The drive uses an adaptive algorithm that accounts for the buffer segment size, the command transfer length, and data transfer rate to determine the optimal point to reconnect.

[5] The BUFFER EMPTY RATIO indicates, on WRITE commands, how empty the drive's buffer shall be before attempting a reselection. This value is changeable by a SCSI initiator port, but the value is ignored by the drive. The drive uses an adaptive algorithm that accounts for the buffer segment size, the command transfer length, and data transfer rate to determine the optimal point to reconnect.

[6] The BUS INACTIVITY LIMIT field (bytes 4 & 5) indicates the time, in 100 microsecond increments, the drive is allowed to assert the Busy signal without handshakes until it shall disconnect. The value of ten indicates the drive is allowed to maintain the Busy signal for one millisecond without handshakes. A value of zero indicates that there is no bus inactivity limit. This value is not changeable by the initiator.

[7] The DISCONNECT TIME LIMIT field (bytes 6 and 7) indicates the minimum time, in 100 microsecond increments, the drive shall remain disconnected until it shall attempt to reconnect. A value of zero indicates the drive is allowed to reconnect immediately. For the typical drive, this is likely to always be zero and the changeable code is always zero (see individual drive's Product Manual, Volume 1).

[8] The CONNECT TIME LIMIT field (bytes 8 and 9) indicates the maximum time in 100 microsecond increments that the target should remain connected until it attempts to disconnect. A setting of zero indicates that the drive is allowed to remain connected indefinitely until it attempts disconnection.

[9] The MAXIMUM BURST SIZE field indicates the maximum amount of data that the drive shall transfer during a data phase before disconnecting if the initiator has granted the disconnect privilege. This value is expressed in increments of 512 bytes (e.g., a value of one means 512 bytes, two means 1024 bytes, etc.). A value of zero indicates there is no limit on the amount of data transferred per connection.

[10] The Data Transfer Disconnect Control (DTDC) field is intended to define further restrictions on when a disconnect is permitted. The various DTDC functions called for by the DTDC Field Codes are given in the table following.

Data Transfer Disconnect Control

DTDC	**Description**
000b	Data transfer disconnect control is not used. Disconnect is controlled by the other fields in this page.
001b	A drive shall not attempt to 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 drive shall not attempt to 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.

If DTDC is nonzero and the maximum burst size is nonzero, the drive shall return CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code set to Illegal Field In Parameter List.

Not all Seagate drive models implement this field. See individual drive’s Product Manual, Volume 1, which indicates if a particular drive implements the DTDC function.

[11] The Enable MODIFY DATA POINTERS (EMDP) bit indicates whether or not the initiator allows the MODIFY DATA POINTERS message to be sent by the target. If the EMDP bit is zero, the target shall not issue the MODIFY DATA POINTERS Message. If the EMDP bit is one, the target is allowed to issue MODIFY DATA POINTERS Message.

[12] The optional Disconnect Immediate (DImm) bit of zero indicates that the drive may disconnect after command phase if it chooses to do so based on its internal algorithms, the setting of the DiscPriv bit in the IDENTIFY message, and the settings of the other parameters in this mode page.

An optional Disconnect Immediate bit of one indicates that the drive shall attempt to disconnect immediately after every command phase for those connections in which disconnections are allowed.

Host adapters in untagged command environments which have relatively high overhead to handle a DISCONNECT message sequence may want to use the Disconnect Immediate feature. However, not all Seagate drive models implement this feature. See individual drive’s Product Manual, Volume 1.

[13] The Fair Arbitration field indicates whether the target should use fair or unfair arbitration when requesting an interconnect tenancy. The field may be used to indicate different fairness methods as specified in the individual protocol documents. See Section 3.4.

[14] The First Burst Size field indicates the maximum amount of data that a SCSI target port may transfer for a command during the same interconnect tenancy in which it receives the command. This value is expressed in increments of 512 bytes (e.g., a value of one means 512 bytes, two means 1024 bytes, etc.). A value of zero indicates that there is no first burst size limit.

8.13.5 Format Parameters page (03h)

The Format Parameters page implementation is defined in Table 158. This table summarizes the function and defines the default or changeability status for each bit. The actual implementation of reserving spare areas for defect management takes place during the FORMAT UNIT command.

Note. In Table 158, zone refers to defect management zone (one or more tracks), not a ZBR (variable track capacity recording) zone. ZBR zones are referred to as notches (page 0Ch is the Notch page).

Table 158. Format Parameters page (03h)

Bit Byte	7	6	5	4	3	2	1	0
Page Descriptor Header								
0	1	0	0	0	0	0	1	1
	PS [1]		PAGE CODE (03h)					
1	PAGE LENGTH (16h)							
Format Parameters [1]								
2,3 Default	TRACKS PER ZONE [2]							
Changeable	[12]							
4,5 Default	ALTERNATE SECTORS PER ZONE [3]							
Changeable	[12]							
6,7 Default	ALTERNATE TRACKS PER ZONE [4]							
Changeable	[12]							
8,9 Default	ALTERNATE TRACKS PER VOLUME [5]							
Changeable	[12]							
10,11 Default	SECTORS PER TRACK [6]							
Changeable	[12]							
12,13 Default	DATA BYTES PER PHYSICAL SECTOR [7]							
Changeable	[12]							
14,15 Default	INTERLEAVE [8]							

Bit Byte	7	6	5	4	3	2	1	0
Changeable	[12]							
16,17 Default	TRACK SKEW FACTOR [9]							
Changeable	[12]							
18,19 Default	CYLINDER SKEW FACTOR [10]							
Changeable	[12]							
20 Default	SSEC [11]	HSEC [11]	RMB [11]	SURF [11]	DRIVE TYPE [11]			
Changeable	[12]							
21,22,23 Default	Reserved							
Changeable	[12]							

[1] The only time this page of parameters may be sent is immediately before sending a FORMAT UNIT command to the drive. The Current parameters for this page are updated immediately but any changes between these Current parameters and the existing media format are not in effect until after the FORMAT UNIT command is completed. A PS bit of 1 indicates this page is savable. The PS bit is not used with the MODE SELECT command.

[2] The TRACKS PER ZONE field indicates the number of tracks the drive allocates to each defect management zone. A zone can be one or more tracks or one or more cylinders. See individual drive's Product Manual, Volume 1, for number of tracks allocated to each defect management zone for that drive model.

[3] The ALTERNATE SECTORS PER ZONE field indicates the number of spare sectors to be reserved for the defined defect management zone. A value of zero indicates that no sectors are to be reserved in each zone for defect management. This is to accommodate hosts that want to manage the defects themselves.

[4] The ALTERNATE TRACKS PER ZONE field indicates the number of spare tracks to be reserved at the end of each defect management zone. A value of zero indicates that no spare tracks are to be reserved in each zone for defect management by the drive.

[5] The ALTERNATE TRACKS PER VOLUME field indicates the number of spare tracks to be reserved at the end of the drive volume. The drive uses these locations for replacing defective sectors. A value of zero indicates that no spare tracks are to be reserved at the end of the unit for defect management. The initiator may change this value for a number between 0 and 255 that is a multiple of the total number of Data Read/Write heads installed. However, it is not changeable on some products.

[6] The SECTORS PER TRACK field indicates the average number of physical sectors the drive has per disc track. This value depends on the selected sector size and ZBR zones. The number of user accessible sectors per track may be fewer than the reported value, since sectors per Track includes sectors set aside for defect management. This value cannot be used to calculate drive user accessible capacity.

Note. The value cannot be directly selected with the MODE SELECT command, but is a report of how the drive is configured.

[7] The DATA BYTES PER PHYSICAL SECTOR field indicates the number of data bytes the drive shall allocate per physical sector. This value equals the block length reported in the MODE SENSE block descrip-

tor. The bytes per physical sector is not directly changeable by the initiator and is not verified on a MODE SELECT command.

[8] The INTERLEAVE field is the interleave value sent to the drive during the last FORMAT UNIT command.

Note. This field is valid only for MODE SENSE commands. The drive ignores this field during MODE SELECT commands.

[9] The TRACK SKEW FACTOR field indicates the average number of physical sectors between the last logical block on one track and the first logical block on the next sequential track of the same cylinder. A value of zero indicates no skew.

Note. This value is not changeable by a SCSI initiator port.

[10] The CYLINDER SKEW FACTOR field indicates the average number of physical sectors between the last logical block of one cylinder and the first logical block of the next cylinder. A value of zero indicates no skew. Cylinder skew will be utilized by a drive but is not changeable by a SCSI initiator port.

[11] The DRIVE TYPE field bits are defined as follows:

The Hard Sectoring (HSEC) bit (bit 6) set to one indicates the drive shall use hard sector formatting.

Bits 0-5, and 7 are not implemented by the drive and are always zero. All bits (0-7) are not changeable.

[12] See individual drive’s Product Manual, Volume 1, MODE SENSE Data section for changeable values.

8.13.6 Rigid Drive Geometry Parameters page (04h)

The Rigid Drive Geometry Parameters page implementation is defined in Table 159. This table summarizes the function and defines the default value

Table 159. Rigid Drive Geometry Parameters page (04h)

Bit Byte	7	6	5	4	3	2	1	0
Page Descriptor Header								
0	1	0	0	0	0	1	0	0
	PS [1]		PAGE CODE (04h)					
1	PAGE LENGTH (16h)							
Rigid Drive Geometry Parameters								
2 Default	NUMBER OF CYLINDERS (MSB) [2]							
Changeable	[8]							
3 Default	NUMBER OF CYLINDERS [2]							
Changeable	[8]							
4 Default	NUMBER OF CYLINDERS (LSB) [2]							
Changeable	[8]							
5 Default	NUMBER OF HEADS [3]							
Changeable	[8]							
6,7,8 Default	STARTING CYLINDER—WRITE PRECOMP [4]							
Changeable	[8]							
9,10,11 Default	STARTING CYLINDER—REDUCED WRITE CURRENT [4]							
Changeable	[8]							
12,13 Default	DRIVE STEP RATE [4]							
Changeable	[8]							
14,15,16 Default	LANDING ZONE CYLINDER [4]							

Table 159. Rigid Drive Geometry Parameters page (04h) (continued)

Bit Byte	7	6	5	4	3	2	1	0
Changeable	[8]							
17 Default	0	0	0	0	0	0	RPL [5]	
Changeable	[8]							
18 Default	ROTATIONAL OFFSET XXh [6]							
Changeable	[8]							
19 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[8]							
20	(MSB)	MEDIUM ROTATION RATE [7]						
21								(LSB)
Changeable	[8]							
22 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[8]							
23 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[8]							

[1] The Parameter Savable (PS) bit of 1 indicates that page 04 parameter data is savable and is saved when a Format Function is performed. In some drives an exception exists that applies to bytes 17 and 18. In the exception drives, bytes 17 and 18 are only saved if the SMP bit in the MODE SELECT command (Table 145) is 1. See individual drive’s Product Manual, Volume 1, “SCSI bus conditions and miscellaneous features supported.” This PS bit is not applicable to the MODE SELECT command.

[2] The NUMBER OF CYLINDERS field defines the number of physical cylinders used for data storage. This may or may not include spare cylinders set aside for flaw reallocation. See individual drive’s Product Manual, Volume 1, which specifies what the drive reports.

The drive uses some additional cylinders for storing drive parameters, defect lists, or for diagnostic purposes. These are not accessible by the user.

[3] The NUMBER OF HEADS field indicates the maximum number of data (read/write) heads on the drive.

[4] Not applicable.

[5] Used for Spindle Synchronization (rotational position locking).

Rotational Position Locking

RPL	**Description**
00b	Indicates that spindle synchronization is automatic. (Automatic master arbitration is used to determine which device in the chain is to be master.)
01b	The target operates as a synchronized-spindle slave.
10b	The target operates as a synchronized-spindle master.
11b	The target operates as a synchronized-spindle master control (not supported by drive).

[6] Rotational skew in the lagging direction used for spindle synchronization. The value XXh given represents a XXh/FFh fractional part of a revolution lagging offset. One revolution lag is maximum. See section on Synchronous Spindle Operation in individual drive’s Product Manual, Volume 1.

[7] On MODE SENSE command, these bytes return drive nominal rotation rate in revolutions per minute for synchronous spindle operation. The bytes have no meaning for MODE SELECT.

[8] See individual drive’s Product Manual, Volume 1, MODE SENSE Data section, for changeable values.

8.13.7 Verify Error Recovery page (07h)

The Verify Error Recovery page (Table 160) specifies the error recovery parameters the target shall use during the VERIFY command and the VERIFY operation of the WRITE AND VERIFY command.

Table 160. Verify Error Recovery page (07h)

Bit **Byte**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Page Descriptor Header								
0	1 PS [1]	0 Reserved	PAGE CODE (07h)					
1	PAGE LENGTH (0Ah)							
Verify Error Recovery Parameters								
2 Default	0	0	0	0	EER [2]	PER [3]	DTE [4]	DCR [5]
	Reserved							
Changeable	[9]							
3 Default	VERIFY RETRY COUNT [6]							
Changeable	[9]							
4 Default	VERIFY CORRECTION SPAN (bits) [7]							
Changeable	[9]							
5 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[9]							
6 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[9]							
7 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[9]							
8 Default	0	0	0	0	0	0	0	0
	Reserved							

Bit Byte	7	6	5	4	3	2	1	0
Changeable	[9]							
9 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[9]							
10 Default	1	1	1	1	1	1	1	1
	(MSB)	VERIFY RECOVERY TIME LIMIT [8]						
Changeable	[9]							
11 Default	1	1	1	1	1	1	1	1
	VERIFY RECOVERY TIME LIMIT [8]							(LSB)
Changeable	[9]							

[1] The Parameter Savable (PS) bit is used only with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the target is capable of saving the page in a non-volatile vendor-specific location.

[2] The Enable Early Recovery (EER) bit, when set to one, allows the drive to apply on-the-fly T>1 ECC correction as soon as possible, before attempting other retry mechanisms. SEEK error retries and message system errors are not affected by this bit. When this bit is set, the DCR bit must be zero. When the EER bit is set to zero, the drive shall apply ECC correction before other retry mechanisms, but shall not perform T>1 corrections on the fly; any successful correction will be reported to the host as a recovered error.

[3] The Post Error (PER) bit, when set to one, indicates the drive reports CHECK CONDITION status and appropriate sense key for any recovered errors encountered. Reporting of unrecoverable errors has priority over reporting of recoverable errors. When set to zero, any errors recovered within the limits established by the other Error Recovery Flags are not reported. Any unrecoverable errors are reported.

[4] The Disable Transfer on Error (DTE) bit is valid only when the PER bit is set to one. When the DTE bit is set to one, it indicates the drive terminates data transfer even for recoverable errors (the drive will transfer the data for the recovered error before terminating the transfer). When DTE is set to zero, data transfer continues if recoverable errors are encountered. If the PER bit is one and the DTE bit is zero, recoverable errors are reported after all data has been transferred.

[5] The Disable Correction (DCR) bit, when set to one, indicates that only on-the-fly ECC correction shall be applied. The exact behavior is product specific. When set to zero, maximum ECC correction shall be applied if correction is possible.

[6] The VERIFY RETRY COUNT sets up the maximum amount of error recovery effort to be applied for each LBA that could not be recovered during a VERIFY operation. The hex value in this field specifies the maximum error recovery level that the drive applies during a VERIFY operation to the recovery of an LBA needing recovery effort. Each level may consist of multiple error recovery steps. See individual drive's Product Manual, Volume 1, for more details on the levels of error recovery available.

[7] The VERIFY CORRECTION SPAN field specifies the size (in bits) of the largest burst data error for which data error correction may be attempted. If the drive does not implement this field, a value of zero is returned in MODE SENSE data.

[8] The VERIFY RECOVERY TIME LIMIT field (bytes 10 and 11) specifies the maximum time in milliseconds that the host allows the drive to spend in error recovery efforts during the execution of a command. The Verify Retry count can also be set to limit the amount of time the drive spends in error recovery of individual LBAs. The total of all times used to recover individual LBAs in the block called for by a command cannot exceed the VERIFY RECOVERY TIME LIMIT value in bytes 10 and 11. Once the drive has reached the error recovery time limit for a particular command, the command ends with a CHECK CONDITION status and an unrecovered error is reported. A RECOVERY TIME LIMIT of FFFFh or 0000h means that the command recovery time is unlimited. A value of 0001h means that no time shall be spent in error recovery. A changeable RECOVERY TIME LIMIT is not supported on all drives supported by this manual. See individual drive's Product Manual, Volume 1, Mode page changeable bit settings for Mode page 01h, bytes 10 and 11.

To disable all types of correction and retries, the initiator sets the EER bit to zero, the PER, DTE, and DCR bits to one, and the number of retries and recovery time limit to zero.

[9] See individual drive's Product Manual, Volume 1, MODE SENSE Data section, for changeable values.

8.13.8 Caching Parameters page (08h)

The Caching Parameters page for MODE SENSE/MODE SELECT defines the parameters that affect the use of the cache (Table 161).

Table 161. Caching Parameters page (08h)

Bit Byte	7	6	5	4	3	2	1	0
Page Descriptor Header								
0	PS [1]	Reserved	PAGE CODE (08h)					
1	PAGE LENGTH (12h)							
2	IC [3]	ABPF [4]	CAP [5]	DISC [6]	SIZE [7]	WCE [8]	MF [9]	RCD [10]
Changeable	[2]							
3	DEMAND READ RETENTION PRIORITY [11]				WRITE RETENTION PRIORITY [12]			
Changeable	[2]							
4	(MSB)	DISABLE PREFETCH TRANSFER LENGTH [13]						
5								(LSB)
Changeable	[2]							
6	(MSB)	MINIMUM PREFETCH [14]						
7								(LSB)
Changeable	[2]							
8	(MSB)	MAXIMUM PREFETCH [15]						
9								(LSB)
Changeable	[2]							
10	(MSB)	MAXIMUM PREFETCH CEILING [16]						
11								(LSB)
Changeable	[2]							
12	FSW [17]	LBCSS [18]	DRA [19]	Reserved				
Changeable	[2]							
13	NUMBER OF CACHE SEGMENTS [20]							

Bit Byte	7	6	5	4	3	2	1	0
Changeable	[2]							
14	(MSB)	CACHE SEGMENT SIZE [21]						
15								(LSB)
Changeable	[2]							
16	Reserved							
17	(MSB)	NON-CACHE SEGMENT SIZE [22]						
18								
19								(LSB)
Changeable	[2]							

[1] The returned Parameter Savable (PS) bit of 1 indicates that page 01h parameter data is savable.

[2] A value of zero means this bit function is not directly changeable by a SCSI initiator port, a value of 1 means the bit function is directly changeable by a SCSI initiator port (see MODE SELECT command). See individual drive's Product Manual, Volume 1, section showing changeable values.

[3] When the Initiator Control (IC) enable bit is set to one, adaptive read look-ahead (ARLA) is disabled. When IC is set to zero, ARLA is enabled. Since Seagate drives covered by this manual never organize the cache according to size of segment, but rather by number of segments, this bit is used to enable or disable ARLA (in non-Seagate equipment, this might be used to designate cache size).

Note. ARLA cannot be disabled in some Seagate drives using the ASAII code. See individual drive's Product Manual, Volume 1.

[4] The ABORT PREFETCH (ABPF) bit, when set to one, with the DRA bit equal to zero, requests that the SCSI device abort the PREFETCH upon selection. The ABPF set to one takes precedence over the Minimum PREFETCH bytes. When set to zero, with the DRA bit equal to zero, the termination of any active PREFETCH is dependent upon Caching Page bytes 4 through 11 and is operation and/or vendor-specific.

[5] The Caching Analysis Permitted (CAP) bit, when set to one, enables caching analysis. A zero indicates caching analysis is disabled. Caching analysis results are placed in the SCSI logging information table, (see Table 122). See individual drive's Product Manual, Volume 1, SCSI Bus Conditions and Miscellaneous Features Supported table.

[6] The Discontinuity (DISC) bit, when set to one, requests that the SCSI device continue the PREFETCH across time discontinuities, such as across cylinders or tracks up to the limits of the buffer, or segment, space available for PREFETCH. When set to zero, the DISC requests that prefetches be truncated at time discontinuities.

[7] The Size Enable (SIZE) bit, when set to one, indicates that the Cache Segment Size is to be used to control caching segmentation. When SIZE equals zero, the Initiator requests that the Number of Cache Segments is to be used to control caching segmentation. For Seagate drives covered by this manual, SIZE is always zero.

[8] Write Cache Enable (WCE).

0: SCSI WRITE commands may not return status and completion message bytes until all data has been written to the media.

1: SCSI WRITE commands may return status and completion message bytes as soon as all data has been received from the host.

[9] Multiplication Factor (MF).

0: The Minimum PREFETCH and Maximum PREFETCH fields are interpreted as a number of logical blocks.

1: Specifies that the target shall interpret the minimum and maximum PREFETCH fields to be specified in terms of a scalar number which, when multiplied by the number of logical blocks to be transferred for the current command, yields the number of logical blocks for each of the respective types of PREFETCH.

[10] READ Cache Disable (RCD).

0: SCSI READ commands may access the cache or the media.

1: SCSI READ commands must access the media. Data cannot come from the cache.

[11] DEMAND READ RETENTION PRIORITY. The cache replacement algorithm does not distinguish between retention in the cache of host-requested data and PREFETCH data. Therefore, this half byte is always 0.

[12] WRITE RETENTION PRIORITY. The cache replacement algorithm does distinguish between retention in the cache of host-requested data and PREFETCH data. Therefore, this half byte is always 0.

[13] DISABLE PREFETCH TRANSFER LENGTH. PREFETCH is disabled for any SCSI READ command whose requested transfer length exceeds this value.

[14] MINIMUM PREFETCH specifies the minimum number sectors to prefetch, regardless of the delay it may cause to other commands.

[15] MAXIMUM PREFETCH specifies the maximum number of logical blocks that may be prefetched. The PREFETCH operation may be aborted before the MAXIMUM PREFETCH value is reached, but only if the MINIMUM PREFETCH value has been satisfied.

[16] MAXIMUM PREFETCH Ceiling specifies an upper limit on the number of logical blocks computed as the maximum prefetch. If the MAXIMUM PREFETCH value is greater than the MAXIMUM PREFETCH CEILING, the value is truncated to the MAXIMUM PREFETCH CEILING value.

[17] The Force Sequential Write (FSW) bit, when set to one, indicates that multiple block writes are to be transferred over the SCSI bus and written to the media in an ascending, sequential, logical block order. When the FSW bit equals zero, the target is allowed to reorder the sequence of writing addressed logical blocks in order to achieve a faster command completion.

[18] The Logical Block Cache Segment Size (LBCSS) bit is not used at this time.

[19] The Disable READ-Ahead (DRA) bit, when set to one, requests that the target not read into the buffer any logical blocks beyond the addressed logical block(s). When the DRA bit equals zero, the target may continue to read logical blocks into the buffer beyond the addressed logical block(s).

[20] The NUMBER OF CACHE SEGMENTS byte gives the number of segments into which the host requests the drive divide the cache.

[21] The CACHE SEGMENT SIZE field indicates the requested segment size in bytes. This manual assumes that the Cache Segment Size field is valid only when the Size bit is one.

[22] The NON-CACHE SEGMENT SIZE field, when greater than zero, specifies to the target the number of bytes the initiator requests that the target allocate for a buffer function when all other cache segments are occupied by data to be retained. If the number is at least one, caching functions in the other segments need not be impacted by cache misses to perform the SCSI buffer function. The impact of the NON-CACHE SEGMENT SIZE equal 0 or the same of this field plus the CACHE SEGMENT SIZE greater than the buffer size is vendor-specific.

In addition to the caching control provided by the Caching Mode page, some 10-byte commands contain control bits DPO and FUA the intent of which is to override the cache page control bits.

DPO: Disable Page Out.

If DPO = 1, the cache replacement algorithm should not replace existing cache data with the current data (if possible). If a cache segment must be overwritten, it should be made the LRU.

FUA: Force Unit Access.

READ: If FUA = 1, the requested data must be read from the media. If cache data which overlaps the request has not yet been written to the media, it should be written before the read is allowed to occur.

WRITE: If FUA = 1, all data must be written to the media before the SCSI operation returns the status and completion message bytes.

8.13.9 Control Mode page (0Ah)

The MODE SENSE/MODE SELECT Control Mode page (Table 162) provides controls over several SCSI features which are applicable to all device types such as tagged queuing, extended contingent allegiance, asynchronous event notification, and error logging.

Table 162. Control Mode page (0Ah)

Bit Byte	7	6	5	4	3	2	1	0
0	PS [1]	Reserved	PAGE CODE (0Ah)					
1	PAGE LENGTH (0Ah)							
2 Default	TST [2]			Reserved			GLTSD [3]	RLEC [4]
Changeable	[7]							
3 Default	QUEUE ALGORITHM MODIFIER [5]				Reserved	QERR [6]		DQUE [8]
Changeable	[7]							
4 Default	Reserved	RAC [9]	Reserved		SWP [10]	RAERP [11]	UAAERP [12]	EAERP [13]
Changeable	[7]							
5 Default	Reserved					AUTOLOAD MODE [14]		
Changeable	[7]							
6 & 7 Default	(MSB)	READY AER HOLDOFF PERIOD [15]						(LSB)
Changeable	[7]							
8 & 9 Default	(MSB)	BUSY TIMEOUT PERIOD [16]						(LSB)
Changeable	[7]							
10	(MSB)	EXTENDED SELF-TEST COMPLETION TIME [17]						
11								(LSB)
Changeable	[7]							

[1] The Parameter Savable (PS) bit of 1 indicates that the page 0Ah parameter data is savable in non-volatile memory.

[2] A Task Set Type (TST) field specifies the type of task set (see Table 163). If the device maintains mode pages per initiator, the TST field, if changeable, shall reflect in all initiator pages the state selected by the most recent MODE SELECT. If the most recent MODE SELECT changes the setting of this field the target shall establish a unit attention condition for all initiators except the one that issued the MODE SELECT command (see SAM-2). The target shall set the additional sense code to Mode Parameters Changed.

Table 163: Task Set types

Value	Description
000b	Task set per logical unit for all initiators
001b	Task set per initiator per logical unit
010b - 111b	Reserved

[3] A Global Logging Target Save Disable (GLTSD) bit of zero allows the target to provide a SCSI target port-defined method for saving log parameters (logged to disk). A GLTSD bit of one indicates that either the target has disabled the target-defined method for saving log parameters or when set by the initiator specifies that the target-defined method shall be disabled.

[4] A Report Log Exception Condition (RLEC) bit of one specifies that the target shall report log exception conditions. A RLEC bit of zero specifies that the drive does not report Log exception conditions.

[5] The QUEUE ALGORITHM MODIFIER field specifies restrictions on the algorithm used for re-ordering commands that are tagged with the SIMPLE QUEUE TAG message.

Table 164: Queue Algorithm Modifier

Value	Definition
0h	Guaranteed data integrity: Requires the device to order the actual execution sequence of the queued command such that data integrity is guaranteed at any time. This requires that if the data transmission of a command was halted at any time, the final value of all data must have exactly the same value it would have had if the command had been executed without queuing. The guaranteed data integrity value (0h) of the queue algorithm modifier bits is the usual default value.
1h	Unrestricted re-ordering allowed: Allows the device to order the actual execution sequence of the queued commands in any manner it selects. Any data integrity problems related to command sequence ordering are explicitly handled by the host operating system software.
2h - 7h	Reserved.
8h - Fh	vendor-specific.

[6] A QUEUE ERROR MANAGEMENT (QERR) bit of zero indicates that those commands still queued after the device has entered the contingent allegiance condition continue execution in a normal manner when that condition has terminated. A QERR bit of one indicates that those commands still queued after the device has entered the contingent allegiance condition shall be aborted when that condition has terminated. A unit attention condition is created for each initiator that had commands in the queue, but not for the initiator detecting the original error.

[7] See individual drive’s Product Manual, Volume 1, MODE SENSE Data section for changeable values.

[8] A Disable Queuing (DQUE) bit of one indicates that tagged queuing is disabled on the drive. Any pending commands in the queue for that I T X nexus is aborted. Any subsequent queue tag message received shall be rejected with a MESSAGE REJECT message and the I/O process shall be executed as an untagged command. A DQUE bit of zero indicates that tagged queuing is enabled, if the drive supports tagged Queuing.

[9] Report A Check (RAC). Not used at this time.

[10] A Software Write Protect (SWP) bit of one specifies that the logical unit shall inhibit writing to the medium after writing all cached or buffered write data, if any. When SWP is one, all commands requiring writes to

the medium shall return CHECK CONDITION status and shall set the sense key to Data Protect and the additional sense code to Write Protected. When SWP is one and the device model defines a write protect (WP) bit in the DEVICE-SPECIFIC PARAMETER field in the mode parameter header (see Table 153), the WP bit shall be set to one for subsequent MODE SENSE commands. A SWP bit of zero specifies that the logical unit may allow writing to the medium, depending on other write inhibit mechanisms implemented by the logical unit. When the SWP bit is zero, the value of the WP bit (if defined) is device model specific. The individual drive's Product Manual, Volume 1, must be consulted to see how each device handles write protect issues. A jumper on the drive option select head usually provides the device model-specific way to control write protect.

[11] The RAERP, UAAERP, and EAERP bits enable specific events to be reported via the asynchronous event reporting protocol. When all three bits are zero, the target shall not use asynchronous event reporting. AER is defined in SAM-2.

A Ready AER Permission (RAERP) bit of one specifies that the target may issue an asynchronous event report upon completing its initialization sequence instead of generating a unit attention condition. A RAERP bit of zero specifies that the target shall not issue an asynchronous event report upon completing its initialization sequence.

Note. If the target's default value for the RAERP bit is one and it does not implement saved parameters or include a hardware switch, then it may be impossible to disable the initialization sequence asynchronous event reporting.

[12] A UNIT ATTENTION AER Permission (UAAERP) bit of one specifies that the target may issue an asynchronous event report instead of creating a unit attention condition upon detecting an unit attention condition event (other than upon completing an initialization sequence). A UAAERP bit of zero specifies that the target shall not issue an asynchronous event reporting instead of creating a unit attention condition.

[13] An Error AER Permission (EAERP) bit of one specifies that the target may issue an asynchronous event report upon detecting a deferred error condition instead of waiting to report the deferred error on the next command. An EAERP bit of zero specifies that the target shall not report deferred error conditions via an asynchronous event reporting.

[14] The Autoload Mode field specifies the action to be taken by a removable medium target when a medium is inserted. For devices other than removable medium devices, this field is reserved. Table 165 shows the usage of the Autoload Mode field.

Table 165: Autoload Mode field

Value	Description
000b	Medium shall be loaded for full access.
001b	Medium shall be loaded for medium auxiliary memory access only.
010b	Medium shall not be loaded.
011b - 111b	Reserved

[15] The READY AER HOLDOFF PERIOD field specifies the minimum time in milliseconds after the target starts its initialization sequence that it shall delay before attempting to issue an asynchronous event report. This value may be rounded up as defined in Section 7.8.

[16] The BUSY TIMEOUT PERIOD field specifies the maximum time, in 100 milliseconds increments, that the initiator allows for the target to remain busy for unanticipated conditions which are not a routine part of commands from the initiator. This value may be rounded down as defined in Section 7.8. A 0000h value in this field is undefined by this manual. An FFFFh value in this field is defined as an unlimited period.

[17] The EXTENDED SELF-TEST COMPLETION TIME field contains advisory data that an initiator may use to determine the time in seconds that the target requires to complete an extended self-test when the target is not interrupted by subsequent commands and no errors occur during execution of the self-test. The initiator should expect this time to increase significantly if other commands are sent to the logical unit while a self-test is in progress or if errors occur during execution of the self-test. targets supporting SELF-TEST CODE field values other than 000b for the SEND DIAGNOSTIC command (see Section 8.47), shall support the EXTENDED SELF-TEST COMPLETION TIME field.

8.13.10 Notch page (0Ch)

The Notch page (Table 166) contains parameters for direct access devices that implement a variable number of blocks per cylinder and support this page. Each section of the drive with a different number of blocks per cylinder is referred to as a notch.

Table 166. Notch page (0Ch)

Bit Byte	7	6	5	4	3	2	1	0
0	PS [1]	0	0	0	1	1	0	0
		Reserved	PAGE CODE (0Ch)					
1	PAGE LENGTH (16h)							
2 Default	ND [2]	LPN [3]	0	0	0	0	0	0
			Reserved					
Changeable	[9]							
3 Default	0	0	0	0	0	0	0	0
	Reserved							
Changeable	[9]							
4,5 Default	(MSB)	MAXIMUM NUMBER OF NOTCHES [4]						(LSB)
Changeable	[9]							
6,7 Default	(MSB)	ACTIVE NOTCH [5]						(LSB)
Changeable	[9]							
8-11 Default	(MSB)	STARTING BOUNDARY [6]						(LSB)
Changeable	[9]							
12-15 Default	(MSB)	ENDING BOUNDARY [7]						(LSB)
Changeable	[9]							
16-23 Default	(MSB)	PAGES NOTCHED [8]						(LSB)
Changeable	[9]							

[1] The Parameter Savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one indicates that the drive is capable of saving the page in a non-volatile vendor-specific location.

[2] A Notched Drive (ND) bit of zero indicates that the device is not notched and that all other parameters in this page shall be returned as zero by the drive. An ND bit of one indicates that the drive is notched. For each supported active notch value, this page defines the starting and ending boundaries of the notch.

[3] A Logical or Physical Notch (LPN) bit of zero indicates that the notch boundaries are based on the physical parameters of the drive. The cylinder is considered most significant, the head least significant. An LPN bit of one indicates that the notch boundaries are based on logical blocks on the drive.

[4] The MAXIMUM NUMBER OF NOTCHES field indicates the maximum number of notches supported by the drive.

[5] The ACTIVE NOTCH field indicates the notch to which this and subsequent MODE SELECT and MODE SENSE commands shall refer, until the active notch is changed by a later MODE SELECT command. The value of the active notch shall be greater than or equal to 0 and less than or equal to the maximum number of notches. An active notch value of zero indicates that this and subsequent MODE SELECT and MODE SENSE commands refer to the parameters that apply across all notches.

[6] The STARTING BOUNDARY field indicates the beginning of the active notch or, if the active notch is zero, the beginning boundary of the drive. If the LPN bit is one, then the four bytes represent a LOGICAL BLOCK ADDRESS. If the LPN bit is zero, then the three most significant bytes shall represent the cylinder number and the least significant byte shall represent the head number. When used with the MODE SELECT command, this field is ignored.

[7] The ENDING BOUNDARY field indicates the ending of the active notch or, if the active notch is zero, the ending of the drive. If the LPN bit is one, then the four bytes represent LOGICAL BLOCK ADDRESS. If the LPN bit is zero, then the three most significant bytes shall represent the cylinder number and the least significant byte shall represent the head number. When used with the MODE SELECT command, this field is ignored.

Each notch shall span a set of consecutive logical blocks on the drive, the notches shall not overlap, and no logical block shall be excluded from a notch.

[8] The PAGES NOTCHED field is a bit map of the mode page codes that indicates which pages contain parameters that may be different for different notches. The most significant bit of this field corresponds to page code 3Fh and the least significant bit corresponds to page code 00h. If a bit is one, then the corresponding mode page contains parameters that may be different for different notches. If a bit is zero, then the corresponding mode page contains parameters that are constant for all notches.

[9] See individual drive’s Product Manual, Volume 1, MODE SENSE Data section, for a table showing codes that indicate which of these bits are changeable by the host using the MODE SELECT command.

8.13.11 Power Condition page (1Ah)

The Power Condition page provides the initiator the means to control the length of time a logical unit will delay before changing its power requirements. There is no notification to the initiator that a logical unit has entered into one of the power conditions.

On the receipt of a command, the device shall adjust itself to the power condition which allows the command to execute. The timer which maps to this power condition and any lower power condition timers shall be reset on receipt of the command. The timer associated with this power condition shall be restarted when the condition that forces the change in power completes.

Table 167. Power Condition page (1Ah)

Bit Byte	7	6	5	4	3	2	1	0
0	PS	Reserved		PAGE CODE (1Ah)				
1	PAGE LENGTH (0Ah)							
2	Reserved							
3	Reserved						IDLE [1]	STANDBY [2]
Changeable	[5]							
4 5 6 7	(MSB)	IDLE CONDITION TIMER [3]						(LSB)
Changeable	[5]							
8 9 10 11	(MSB)	STANDBY CONDITION TIMER [4]						(LSB)
Changeable	[5]							

[1] An IDLE bit of one indicates a logical unit shall use the IDLE CONDITION TIMER value to determine the length of inactivity time to wait before entering the Idle condition. An IDLE bit of zero indicates a logical unit shall not enter the Idle condition.

[2] A STANDBY bit of one indicates a logical unit shall use the STANDBY CONDITION TIMER value to determine the length of inactivity time to wait before entering the Idle condition. A STANDBY bit of zero indicates a logical unit shall not enter the Idle condition.

[3] The IDLE CONDITION TIMER field indicates the inactivity time in 100 millisecond increments that the logical unit shall wait before entering the Idle condition. Minimum time is 500 milliseconds.

[4] The STANDBY CONDITION TIMER field indicates the inactivity time in 100 millisecond increments that the logical unit shall wait before entering the Standby condition.

[5] See individual drive's Product Manual, Volume 1, MODE SENSE Data section, for a table showing codes that indicate which of these bits are changeable by the host using the MODE SELECT command.

8.13.12 Xor Control Mode page (10h)

The Xor Control Mode page (Table 168) provides the initiator with the means to obtain or modify certain Xor operating parameters of the target.

Table 168. Xor Control Mode page (10h)

Bit Byte	7	6	5	4	3	2	1	0
0	PS [1]	Reserved	PAGE CODE (10h)					
1	PAGE LENGTH (16h)							
2	Reserved						XORDIS [2]	Reserved
3	Reserved							
4 - 7	(MSB)			MAXIMUM XOR WRITE SIZE [3]				
								(LSB)
Changeable	[7]							
8 - 11	(MSB)			Reserved				
								(LSB)
Changeable	[7]							
12 - 15	(MSB)			MAXIMUM REGENERATE SIZE [4]				
								(LSB)
Changeable	[7]							
16 - 19	(MSB)			MAXIMUM REBUILD TRANSFER SIZE [5]				
								(LSB)
Changeable	[7]							
20 21	Reserved							
Changeable	[7]							
22 23	REBUILD DELAY [6]							
Changeable	[7]							

[1] The parameters savable (PS) bit is only used with the MODE SENSE command. This bit is reserved with the MODE SELECT command. A PS bit of one shall indicate that the target is capable of saving the mode page in a non-volatile vendor-specific location.

[2] An Xor Disable (XORDIS) bit of zero enables the Xor operations within a device. An XORDIS bit of one disables the Xor operations within a device. If the XORDIS bit is set to one and an Xor command is sent to the target, the command shall be terminated with CHECK CONDITION status. The sense data shall be set to ILLEGAL REQUEST: INVALID COMMAND OPERATION CODE.

[3] The MAXIMUM XOR WRITE SIZE field specifies the maximum transfer length in blocks that the target accepts for a single XDWRITE, XDWRITE EXTENDED or XPWRITE command.

[4] The MAXIMUM REGENERATE SIZE field specifies the maximum regenerate length in blocks that the target accepts for the REGENERATE command.

[5] The MAXIMUM REBUILD TRANSFER SIZE field specifies the maximum transfer length in blocks that the target shall use for commands issued as a temporary initiator (e.g., READ and XPWRITE commands during a rebuild operation). This field does not limit the rebuild size.

[6] The REBUILD DELAY field specifies the minimum time in milliseconds between successive READ commands during a rebuild operation.

[7] See individual drive's Product Manual, Volume 1, MODE SENSE Data section, for a table showing codes that indicate which of these bits are changeable by the host using the MODE SELECT command.

8.13.13 Vital product data (VPD) parameters

8.13.13.1 VPD parameters overview

This subclause defines the VPD pages used with direct-access block type devices.

The VPD page codes specific to direct-access block devices are defined in Table 169.

Table 169. Direct-access device VDP page codes

VPD page code	VPD page name	Reference	Support requirements
B0h	Block Limits VPD page	Section 8.13.13.2	Optional
B1h - BFh	Reserved for this device type		

8.13.13.2 Block Limits VPD page

The Block Limits VPD page (Table 170) provides the initiator with the means to obtain certain operating parameters of the logical unit.

Table 170. Block Limits VPD page

Bit Byte	7	6	5	4	3	2	1	0
0	PERIPHERAL QUALIFIER [1]			PERIPHERAL DEVICE TYPE [1]				
1	PAGE CODE (B0h)							
2	Reserved							
3	PAGE LENGTH (0Ch)							
4 - 5	Reserved							
6	(MSB)	OPTIMAL TRANSFER LENGTH GRANULARITY [2]						
7								(LSB)
8 - 11	(MSB)	MAXIMUM TRANSFER LENGTH [3]						
								(LSB)
12 - 15	(MSB)	OPTIMAL TRANSFER LENGTH [4]						
								(LSB)

[1] The PERIPHERAL QUALIFIER field and PERIPHERAL DEVICE TYPE field value of 00h indicates a direct-access device (magnetic disc) is connected to this logical unit.

[2] The OPTIMAL TRANSFER LENGTH GRANULARITY field specifies the optimal transfer length granularity in blocks for a single PRE-FETCH, READ, READ LONG, VERIFY, WRITE, WRITE AND VERIFY, WRITE LONG, XDREAD, XDWRITE, XDWRITEREAD, or XPWRITE command. Transfers with lengths not equal to a multiple of this value may incur significant delays in processing.

[3] The MAXIMUM TRANSFER LENGTH field specifies the maximum transfer length in blocks that the target accepts for a single PRE-FETCH, READ, READ LONG, VERIFY, WRITE, WRITE AND VERIFY, WRITE

LONG, XDREAD, XDWRITE, XDWRITEREAD, or XPWRITE command. Requests for transfer lengths exceeding this limit result in CHECK CONDITION status with a sense key of ILLEGAL REQUEST and an additional sense code of INVALID FIELD IN CDB. A value of zero means there is no reported limit on the transfer length.

[4] The OPTIMAL TRANSFER LENGTH field specifies the optimal transfer length in blocks for a single PRE-FETCH, READ, READ LONG, VERIFY, WRITE, WRITE AND VERIFY, WRITE LONG, XDREAD, XDWRITE, XDWRITEREAD, or XPWRITE command.

Transfers with transfer length exceeding this value may incur significant delays in processing.

8.13.14 Port Control Mode page (19h)

The Port Control Mode page contains the parameters that affect SCSI target port operation options. The page shall be implemented by LUN 0 of all SPI SCSI target 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 the ANSI SPC-3 standard.

Each SCSI target port shall maintain an independent set of port control mode page parameters for each SCSI initiator port. The parameters saveable bit in the mode page format header returned with MODE SENSE command shall be set to zero if the long mode page format is being used (i.e., Long bit set to one), indicating the parameters are not saved through resets.

After a MODE SELECT command, parameter settings shall remain in effect until either:

a. settings are changed by another MODE SELECT command,

b. a logical unit reset of LUN 0 occurs,

c. an SDTR negotiation successfully completes,

d. a WDTR negotiation successfully completes, or

e. a PPR negotiation successfully completes with the HOLD_MCS bit set to zero.

Table 171. Port Control Mode page short format (19h)

Bit Byte	7	6	5	4	3	2	1	0
0	PS [1]	Long (0)	PAGE CODE (19h)					
1	PAGE LENGTH (06h)							
2	Reserved				PROTOCOL IDENTIFIER (1h) [2]			
3	Reserved							
4	(MSB)	SYNCHRONOUS TRANSFER TIMEOUT [3]						
5								(LSB)
6	Reserved							
7	Reserved							

[1] The Parameter Savable (PS) bit of 0 indicates that the page 19h parameter data is not savable in non-volatile memory.

[2] The PROTOCOL IDENTIFIER field value of 1h indicates that this mode page applies to a SPI SCSI device. See the SPC-3 standard for other port control page protocol identifiers.

[3] The SYNCHRONOUS TRANSFER TIMEOUT field indicates the maximum amount of time in 1 ms increments that the SCSI target port shall wait before generating an error by doing an unexpected bus free. The SCSI target port 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 ACT 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.

Table 172. Port Control Mode page long format (19h)

Bit Byte	7	6	5	4	3	2	1	0
0	PS [1]	Long (1)	PAGE CODE (19h)					
1	SUBPAGE CODE [2]							
2	(MSB)	PAGE LENGTH (n – 3)						
3								(LSB)
4	Reserved							
5	Reserved				PROTOCOL IDENTIFIER = 1h [3]			
6	PROTOCOL-SPECIFIC MODE PARAMETERS							
n								

[1] The Parameter Savable (PS) bit of 0 indicates that the page 19h parameter data is not savable in non-volatile memory.

[2] The SUBPAGE CODE field indicates which subpage is being accessed. If the SUBPAGE CODE field is zero, the SCSI target device 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.

Subpage Code	Description	Reference Section
1	Margin control subpage	8.13.14.1
2	Saved training configuration values subpage	8.13.14.2
3	Negotiated settings subpage	8.13.14.3
4	Report transfer capabilities subpage	8.13.14.4

[3] The PROTOCOL IDENTIFIER field value of 1h indicates that this mode page applies to a SPI SCSI device. See the SPC-3 standard for other port control page protocol identifiers.

8.13.14.1 Margin control subpage

The margin control subpage contains parameters that set and report margin control values for usage between the SCSI initiator/target port pair on subsequent synchronous and paced transfers.

A MODE SENSE command shall return the current settings for the SCSI initiator/target port pair. Fields that are not implemented shall be reported as zero.

Note. The default value of each margin control field should be 0000b.

The margin control fields indicate absolute conditions centered around their default values. Absolute conditions means that the previous history of the parameter has no relevance to the value of the parameter.

The maximum supported setting for each field is 0111b and the minimum supported setting for each field is 1000b. Up to 16 distinct values are available for each field, representing monotonically changing device response. Devices that support fewer than 16 distinct values for a field should round non-supported settings to a supported value.

The actual response of a SCSI device to a field value is vendor-specific and calibration of the actual minimum and maximum responses to different field values is not defined in this manual or the ANSI SPI-4 specification. Margin control settings should not cause the driver to violate SPI-4 specification electrical limits. Margin control settings should affect only the REQUEST, ACKNOWLEDGE, DATA BUS, P_CRCA, and DB(P1) signals and should affect all of these signals driven by the device by the same amount.

The margin control fields contain two's complement values as shown in the table below.

Value		Parameter values
Binary	Decimal	
0111b	7	Maximum setting
0110b	6	
0101b	5	
0100b	4	
0011b	3	
0010b	2	
0001b	1	
000b	0	Recommended default value
111b	–1	
1110b	–2	
1101b	–3	
1100b	–4	
1011b	–5	
1010b	–6	
1001b	–7	
1000b	–8	Minimum setting

Table 173. Margin control subpage

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	DRIVER STRENGTH [1]				Reserved			
2	DRIVER ASYMMETRY [2]				DRIVER PRECOMPENSATION [3]			
3	DRIVER SLEW RATE [4]				Reserved			
4	Reserved							
5	Reserved							
6	Reserved							
7	Vendor-specific							
8 . . 15	Reserved							

[1] The DRIVER STRENGTH field indicates the relative amount of driver source current used by the driver (see Section 7.3.2 in the ANSI SPI-4 specification). The DRIVER STRENGTH field affect both the strong and weak drivers. A larger value indicates more driver source current.

[2] The DRIVER ASYMMETRY field indicates the relative difference between the amplitudes of asserted and negated signals launched from the driver. A larger value indicates a relatively stronger asserted signal compared to the negated signal.

[3] The DRIVER PRECOMPENSATION field indicates the relative difference between the weak driver and the strong driver amplitudes when pre-compensation is enabled. A larger value indicates a larger difference between the weak and strong amplitudes.

[4] The DRIVER SLEW RATE field indicates the relative difference between the assertion and negation magnitudes divided by the rise or fall time. A larger value indicates a faster slew rate.

8.13.14.2 Saved training configuration values subpage

The saved training configuration values subpage is used to report the SCSI device’s saved training configuration values. These vendor-specific values are maintained by the SCSI device when the retain training information option is enabled (see Section 2.12.8.7). The fields are listed in Table 174 however the content of the field is vendor-specific.

Only values for the current I_T nexus are reported.

Table 174. Saved training configuration values subpage

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
3								
4	MSB	DB(0) Value						
7								LSB
.	.							
.	.							
64	MSB	DB(15) Value						
67								LSB
68	MSB	P_CRCA Value						
71								LSB
72	MSB	P1 Value						
75								LSB
76	MSB	BSY Value						
79								LSB
80	MSB	SEL Value						
83								LSB
84	MSB	RST Value						
87								LSB
88	MSB	REQ Value						
91								LSB
92	MSB	ACK Value						
95								LSB
96	MSB	ATN Value						
99								LSB
100	MSB	C/D Value						
103								LSB
104	MSB	I/O Value						
107								LSB
108	MSB	MSG Value						
111								LSB
112	Reserved							
127								

8.13.14.3 Negotiated settings subpage

The negotiated setting subpage is used to report the negotiated settings of a SCSI target port for the current I_T nexus.

Table 175. Negotiated settings subpage

Bit Byte	7	6	5	4	3	2	1	0
0	TRANSFER PERIOD FACTOR [1]							
1	Reserved							
2	REQ/ACK OFFSET [2]							
3	TRANSFER WIDTH EXPONENT [3]							
4	Reserved	PROTOCOL OPTIONS BITS [4]						
5	Reserved				TRANSCEIVER MODE		SENT PCOMP_EN	RECEIVED PCOM_EN
6	Reserved							
7	Reserved							

[1] The TRANSFER PERIOD FACTOR field indicates the negotiated transfer period factor for the current I_T nexus.

[2] The REQ/ACK OFFSET field indicates the negotiated REQ/ACK offset for the current I_T nexus.

[3] The TRANSFER WIDTH EXPONENT field indicates the negotiated transfer width exponent for the current I_T nexus.

[4] The PROTOCOL OPTIONS BITS field contains the negotiated protocol options for the current I_T nexus.

[5] The RECEIVED PCOMP_EN bit contains the value of the PCOMP_EN bit received by the SCSI target port for the current I_T nexus.

[6] The SENT PCOMP_EN bit contains the value of the PCOMP_EN bit sent by the SCSI target port for the current I_T nexus.

[7] The TRANSCEIVER MODE field specifies the current bus mode of the SCSI target port as defined below.

Code	Bus mode
00b	Unknown (e.g., device not capable of reporting bus mode)
01b	Single-ended
10b	Low Voltage Differential
11b	High Voltage Differential

8.13.14.4 Report transfer capabilities subpage

The Report transfer capabilities subpage is used to report the transfer capabilities for the SCSI target port. The values in this subpage are not changeable via a MODE SELECT command.

Table 176. Report transfer capabilities subpage

Bit Byte	7	6	5	4	3	2	1	0
0	MINIMUM TRANSFER PERIOD FACTOR [1]							
1	Reserved							
2	MAXIMUM REQ/ACK OFFSET [2]							
3	MAXIMUM TRANSFER WIDTH EXPONENT [3]							
4	PROTOCOL OPTIONS BITS SUPPORTED [4]							
5	Reserved							
6	Reserved							
7	Reserved							

[1] The MINIMUM TRANSFER PERIOD FACTOR field shall be set to the smallest value of the transfer period factor supported by the SCSI target port.

[2] The MAXIMUM REQ/ACK OFFSET shall be set to the largest value of the REQ/ACK offset supported by the SCSI target port.

[3] The MAXIMUM TRANSFER WIDTH EXPONENT field shall be set to the largest value of the transfer width exponent supported by the SCSI target port.

[4] The SCSI target port shall set the bits in the PROTOCOL OPTIONS BITS SUPPORTED field to indicate the protocol options supported by the SCSI target port.

8.13.15 Informational Exceptions Control page (1Ch)

The Informational Exceptions Control page (Table 177) defines the methods used by the target to control the reporting and the operations of specific informational exception conditions. This page shall only apply to informational exceptions that report an additional sense code of Failure Prediction Threshold Exceeded to the initiator.

Informational exception conditions occur as the result of vendor-specific events within a SCSI target port. An informational exception condition may occur asynchronous to any commands issued by an initiator.

Mode page 1Ch may be used by the drive to implement the S.M.A.R.T. system. S.M.A.R.T. is an acronym for Self-Monitoring Analysis and Reporting Technology. The intent of the S.M.A.R.T. system is to recognize conditions that indicate imminent drive failure, and provide sufficient warning to the host system of impending failure. Thus the host system may perform data backup before it is too late (see Section 9.1).

Table 177. Informational Exceptions Control page (1Ch)

Bit Byte	7	6	5	4	3	2	1	0
Page Descriptor Header								
0	PS	Reserved	PAGE CODE (1Ch)					
1	PAGE LENGTH (0Ah)							
Informational Exceptions Control Parameters								
2	PERF [1]	Reserved		EWASC [3]	DEXCPT [2]	TEST [4]	Reserved	LOGERR [5]
Changeable	[10]							
3	Reserved				MRIE [6]			
Changeable	[10]							
4 5 6 7	(MSB)	INTERVAL TIMER [7] [9]						(LSB)
Changeable	[10]							
8 9 10 11	(MSB)	REPORT COUNT [8] [9]						(LSB)
Changeable	[10]							

[1] A Performance (PERF) bit of zero indicates that informational exception operations that are the cause of delays are acceptable. A Perf bit of one indicates the target shall not cause delays while doing informational exception operations. A Perf bit set to one may cause the target to disable some or all of the informational exception operations, thereby limiting the reporting of informational exception conditions.

[2] A Disable Exception Control (DEXCPT) bit of zero indicates the failure prediction threshold exceeded reporting shall be enabled. The method of reporting the failure prediction threshold exceeded when the DEXCPT bit is set to zero is determined from the METHOD OF REPORTING INFORMATIONAL EXCEPTIONS field. A DEXCPT bit of one indicates the target shall disable reporting of the failure prediction threshold exceeded. The METHOD OR REPORTING INFORMATIONAL EXCEPTIONS field is ignored when DEXCPT is set to one and EWASC is set to zero.

[3] An Enable Warning (EWASC) bit of zero indicates the target shall disable reporting of the warning. The METHOD OF REPORTING INFORMATIONAL EXCEPTIONS field is ignored when DEXCPT is set to one and EWASC is set to zero. An EWASC bit of one indicates warning reporting shall be enabled. The method for reporting the warning when the EWASC bit is set to one is determined from the METHOD OF REPORTING INFORMATION EXCEPTIONS field.

[4] A TEST bit of one instructs the drive to create false drive failure(s) at the next interval time, provided that the DEXCPT bit is not set. The MRIE and REPORT COUNT fields apply as specified in this document. A false drive failure will be reported as sense code/qualifier 5DFF (FF for false failure versus a true failure 5D00). A TEST bit of zero instructs the drive to not generate any false drive failure notifications.

[5] The Log Errors (LOGERR) bit of zero indicates that the logging of informational exception conditions within a SCSI target port is vendor-specific. A LOGERR bit of one indicates the target shall log informational exception conditions.

[6] The Method of Reporting Informational Exceptions (MRIE) field indicates the methods that shall be used by the target to report informational exception conditions (see Table 178). The priority of reporting multiple information exceptions is vendor-specific.

[7] The INTERVAL TIMER field indicates the period in 100 millisecond increments for reporting that an informational exception condition has occurred. The target shall not report informational exception conditions more frequently than the time specified by the INTERVAL TIMER field and as soon as possible after the timer interval has elapsed. After the informational exception condition has been reported, the interval timer shall be restarted. A value of zero in the INTERVAL TIMER field indicates that the target shall only report the informational exception condition one time. A value of FFFFFFFFh in the INTERVAL TIMER field shall indicate the timer interval is vendor-specific. If the method of reporting informational exceptions field has a value of six, then the INTERVAL TIMER field is ignored.

[8] The REPORT COUNT field indicates the number of times to report an informational exception condition to the initiator. A value of zero in the REPORT COUNT field indicates there is no limit on the number of times the target shall report an informational exception condition.

[9] Maintaining the INTERVAL TIMER and REPORT COUNT after power cycles and/or resets by the target shall be vendor-specific. [Seagate: A predicted failure is preserved through resets and power cycles. The count of how many informational exception conditions that were reported by the target to any initiator is cleared by a reset of power cycle.]

[10] See individual drive’s Product Manual, Volume 1, MODE SENSE Data section, for a table showing codes that indicate which of these bits are changeable by the host using the MODE SELECT command.

Table 178. Format of Method of Reporting Informational Exceptions (MRIE) field

MRIE	Description
0h	No reporting of informational exception condition. This method instructs the target to not report information exception conditions.
1h	Asynchronous event reporting. This method instructs the target to report informational exception conditions by using the rules for asynchronous event reporting as described in the SCSI Architecture Model and the relevant Protocol Standard. The sense key shall be set to Recovered Error and the additional sense code shall indicate the cause of the informational exception condition.
2h	Generate UNIT ATTENTION. This method instructs the target to report informational exception conditions by returning a CHECK CONDITION status on any command. The sense key shall be set to UNIT ATTENTION and the additional sense code shall indicate the cause of the informational exception condition. The command that has the CHECK CONDITION shall not be executed before the informational exception condition is reported.
3h	Conditionally generate recovered error. This method instructs the target to report informational exception conditions, if the reporting of recovered errors is allowed, by returning a CHECK CONDITION status. If the Test bit equals zero, the status may be returned on any command after the informational exception condition occurs. If the Test bit equals one, the status shall be returned on the next command that is normally capable of returning an informational exception condition when the TEST bit equals zero. The sense key shall be set to RECOVERED ERROR and the additional sense code shall indicate the cause of the informational exception condition. The command that has the CHECK CONDITION shall complete without error before any informational exception condition may be reported.
4h	Unconditionally generate recovered error. This method instructs the target to report informational exception conditions, regardless of the value of the per bit of the error recovery mode page, by returning a CHECK CONDITION status. If the TEST bit equals zero, the status may be returned on any command after the informational exception condition occurs. If the TEST bit equals one, the status shall be returned on the next command that is normally capable of returning an informational exception condition when the TEST bit equals zero. The sense key shall be set to RECOVERED ERROR and the additional sense code shall indicate the cause of the informational exception condition. The command that has the CHECK CONDITION shall complete without error before any informational exception condition may be reported.
5h	Generate no sense. This method instructs the target to report informational exception conditions by returning a CHECK CONDITION status. If the TEST bit equals zero, the status may be returned on any command after the informational exception condition occurs. If the TEST bit equals one, the status shall be returned on the next command that is normally capable of returning an informational exception condition when the TEST bit equals zero. The sense key shall be set to NO SENSE and the additional sense code shall indicate the cause of the informational exception condition. The command that has the CHECK CONDITION shall complete without error before any informational exception condition may be reported.
6h	Only report information exception condition on request. This method instructs the target to preserve the informational exception(s) information. To find out about information exception conditions the initiator polls the target by issuing an unsolicited REQUEST SENSE command. The sense key shall be set to NO SENSE and the additional sense code shall indicate the cause of the informational exception condition.
7h - Bh	Reserved.
Ch - Fh	Vendor-specific.

8.14 MODE SENSE (10) command (5Ah)

The MODE SENSE (10) command provides a means for the drive to report drive operating mode parameters to the initiator. It is a complementary command to the MODE SELECT (10) command.

Table 179. MODE SENSE (10) command (5Ah)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	0	1	1	0	1	0
1	RESERVED [1]			LLBAA [2]	DBD [3]	Reserved		
2	PC [4]		PAGE CODE [5]					
3	SUBPAGE CODE [5]							
4 : 6	Reserved							
7	(MSB)	ALLOCATION LENGTH [6]						
8								(LSB)
9	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] If the Long LBA Accepted (LLBAA) bit is one, the target is allowed to return parameter data with the LONGLBA bit equal to one. If the LLBAA bit is zero, the LONGLBA bit shall be zero in the parameter data returned by the target.

[3] A Disable Block Descriptors (DBD) bit of zero indicates that the drive may return zero or more block descriptors in the returned MODE SENSE data, at the drive’s discretion. Seagate SCSI-2 products return one block descriptor if the DBD bit is zero. A DBD bit of one specifies that the drive shall not return any block descriptors in the returned MODE SENSE data. Byte 3 of the MODE SENSE header contains 00h to indicate a block descriptor length of zero.

[4] The content of mode parameter bytes is determined by the value of the PC (Page Control) field bits specified in CDB byte 2, bits 6 and 7. The drive shall return the same PAGE LENGTH for each supported page regardless of the value of PC. The PC field is defined in Table 180.

Table 180: Page Control field bits affect on mode parameters returned

Bit 7	Bit 6	
0	0	Return current values. The current values are the values currently being used by the drive to control its operation. After a POWER ON RESET, a HARD RESET, or a BUS DEVICE RESET message the current values are equal to the saved values (if saved values can be retrieved) or the default values (if saved values cannot be retrieved). The current value of a parameter is updated by a MODE SELECT command if the MODE SELECT command ends with GOOD status returned.
0	1	Return changeable values. The changeable values of any page is a mask that indicates the parameters that shall be changed via a MODE SELECT command and the parameters that shall not. Each returned parameter byte shall contain ones where a field or bit may be changed and zeros where a field or bit may not be changed.
1	0	Return default values. The default values are the values to which the drive sets the current values after a reset condition unless valid saved values are available.
1	1	Return saved values. The saved values are the values the drive stores in nonvolatile memory. The saved values of any changeable parameter can be set to new values via a MODE SELECT command with the SMP bit set to one. For non-changeable parameters, the default value is used.

The Block descriptor contains its normal values regardless of the value of the PC. Unsupported fields or bits within a page are returned as zeros for all PC values.

[5] The PAGE CODE and SUBPAGE CODE fields specify which mode pages and subpages to return.

Page Codes that may be supported by the drive are summarized here (see individual drive's Product Manual, Volume 1):

Page Code	Description
00h	Unit Attention page parameters (returned last of the pages)
01h	Error Recovery parameters
02h	Disconnect/Reconnect Control parameters
03h	Format parameters
04h	Rigid Drive Geometry parameters
07h	Verify Error Recovery page parameters
08h	Caching Parameters page
0Ah	Control Mode page
0Ch	Notch page
1Ah	Power Condition page
10h	Xor Control Mode page
19h	Port Control Mode page
1Ch	Informational Exceptions Control page
3Fh	Return all supported pages

An application client may request any one or all of the supported mode pages from the device server. If an application client issues a MODE SENSE command with a page code or subpage code value not implemented by the target, the device server shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST and the additional sense code to INVALID FIELD IN CDB.

[6] The ALLOCATION LENGTH specifies the number of bytes that the initiator has allocated for returned MODE SENSE data. An ALLOCATION LENGTH of zero indicates that no MODE SENSE data shall be transferred. This condition shall not be considered as an error. Any other value indicates the maximum number of bytes that shall be transferred. The drive shall terminate the DATA IN phase when ALLOCATION LENGTH bytes have been transferred or when all available MODE SENSE data has been transferred to the initiator, whichever is less.

[7] See Table 1, "CONTROL field," in Section 7.2.1.

Table 181. MODE SENSE (10) parameter list

Bit Byte	7	6	5	4	3	2	1	0
0,1	SENSE DATA LENGTH [1]							
2	0	0	0	0	0	0	0	0
	MEDIUM TYPE [2]							
3	WP [3]	0	0	DPO-FUA [8]	0	0	0	0
		Reserved			Reserved			
4,5	Reserved							
6,7	0	0	0	0	1	0	0	0
	BLOCK DESCRIPTOR LENGTH (8 decimal) [4]							

Block Descriptor Data

0	0	0	0	0	0	0	0	0
	DENSITY CODE [5] [6]							
1	0	0	0	0	0	0	0	0
	NUMBER OF BLOCKS (MSB) [6]							
2	0	0	0	0	0	0	0	0
	NUMBER OF BLOCKS [6]							
3	0	0	0	0	0	0	0	0
	NUMBER OF BLOCKS (LSB) [6]							
4	0	0	0	0	0	0	0	0
	Reserved							
5	BLOCK LENGTH (MSB) [7]							
6	BLOCK LENGTH [7]							
7	BLOCK LENGTH (LSB) [7]							

Parameter Information

0-n	MODE SENSE PAGE HEADERS AND THEIR PARAMETERS

[1] The SENSE DATA LENGTH field specifies the length in bytes of the following MODE SENSE data that is available to be transferred during the DATA IN phase. The SENSE DATA LENGTH value does not include itself.

[2] The drive supports only 00h (default medium) in the MEDIUM TYPE field.

[3] A Write Protect (WP) bit of zero indicates the medium is write enabled. A WP bit of one indicates the medium is write protected.

[4] The BLOCK DESCRIPTOR LENGTH specifies the length in bytes of the block descriptor. It is equal to the number of bytes in the block descriptor (8) and does not include the page headers and mode parameters, if any. The drive sends one block descriptor.

Each block descriptor specifies the medium characteristics for all or part of a logical unit. Each block descriptor contains a DENSITY CODE, a NUMBER OF BLOCKS, and a BLOCK LENGTH.

[5] a. Drives that do not support capacity programming have only 00h (default density) in the DENSITY CODE field.

b. Drives that support capacity programming may have a value in this field that states either the logical or actual capacity of the drive.

[6] a. For drives that do not support capacity programming, the NUMBER OF BLOCKS field specifies the number of logical blocks of the medium that meets the DENSITY CODE and BLOCK LENGTH in the block descriptor. A NUMBER OF BLOCKS field value of zero indicates that all of the remaining logical blocks of the logical unit have the medium characteristics specified by the block descriptor.

b. For drives that have capacity programming capability, these bytes do not report back the drive capacity on some models. These bytes are always zero on those models. Some models do report drive capacity in bytes 0, 1, 2, and 3, so those bytes will be nonzero. See individual drive’s Product Manual, Volume 1, for the drive of interest.

[7] The BLOCK LENGTH, as defined after a format function, specifies the length in bytes of each logical block described by the block descriptor. Default is 512 if no MODE SELECT command is received before the format command. The usual valid values are 256 through 4096. Some drive products can format down to 180 bytes per sector. Some products can only format an even numbered value of bytes per sector (180 - 4096).

[8] When used with the MODE SENSE command, a DPO-FUA bit of zero indicates that the target does not contain a cache memory or does not support the DPO and FUA bits. A DPO-FUA bit of one indicates that the target supports the DPO and FUA bits.

8.15 MOVE MEDIUM command (A7h)

Not described in this manual. If received, a CHECK CONDITION status is sent.

8.16 PERSISTENT RESERVE IN command (5Eh)

The PERSISTENT RESERVE IN command (see Table 182) is used to obtain information about persistent reservations and reservation keys that are active within a target. This command is used in conjunction with the PERSISTENT RESERVE OUT command (see Section 8.17).

When a target receives a PERSISTENT RESERVE IN command during the same time RESERVE (10) or RESERVE (6) logical unit or extent reservations or SMC element reservations are active (see sections 8.39 and 8.40), the command shall be rejected with a RESERVATION CONFLICT status. A PERSISTENT RESERVE IN command shall not conflict with any persistent reservation.

The subject of reservations is large and detailed, and therefore not covered entirely in this manual. It is recommended that the reader refer to ANSI document T10/1157D, sections 5.3, 7.11, 7.12, 7.16, 7.17, 7.21, and 7.22, for a greater understanding of and details pertaining to the reservation process as needed for design purposes.

Table 182. PERSISTENT RESERVE IN command (5Eh)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (5Eh)							
1	Reserved			SERVICE ACTION [2]				
2	Reserved							
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	(MSB)	ALLOCATION LENGTH [1]						
8								(LSB)
9	CONTROL [3]							

[1] The actual length of the PERSISTENT RESERVE IN parameter data is available in a parameter data field. The ALLOCATION LENGTH field in the CDB indicates how much space has been reserved for the returned parameter list. If the length is not sufficient to contain the entire parameter list, the first portion of the list shall be returned. This shall not be considered an error. If the remainder of the list is required, the initiator should send a new PERSISTENT RESERVE IN command with an ALLOCATION LENGTH field large enough to contain the entire list.

[2] The SERVICE ACTION codes for the PERSISTENT RESERVE IN command are defined in Table 183.

Table 183: PERSISTENT RESERVE IN Service Action codes

Code	Name	Description
00h	READ Keys	Reads all registered Reservation Keys. The READ KEYS service action requests that the target return a parameter list containing a header and a list of each currently registered I_T nexus’ reservation key. If multiple I_T nexuses have registered with the same key, then that key value shall be listed multiple times, once for each such registration. For more information on READ KEYS see SPC-3, Section 5.5.2.3.2.
01h	READ Reservations	The READ RESERVATIONS service action requests that the target return a parameter list containing a header and the persistent reservations, if any, that are present in the target. Multiple persistent reservations may be returned only if element persistent reservations are present. For more information on READ RESERVATION see SPC-3, Section 5.5.2.3.3.
02h	REPORT Capabilities	Returns capability information. The REPORT CAPABILITIES service action requests that the target return information on persistent reservation features.
03h-1Fh	Reserved	Reserved.

[3] See Table 1, “CONTROL field,” in Section 7.2.1.

8.16.1 PERSISTENT RESERVE IN parameter data for READ Keys

The format for the parameter data provided in response to a PERSISTENT RESERVE IN command with the READ Keys service action is shown in Table 184.

Table 184. PERSISTENT RESERVE IN parameter data for READ Keys

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PRGENERATION [1]						
3								(LSB)
4	(MSB)	ADDITIONAL LENGTH (n–7) [2]						
7								(LSB)
	RESERVATION KEY LIST [3]							
8	(MSB)	FIRST RESERVATION KEY						
15								(LSB)
	. . .							
n–7	(MSB)	LAST RESERVATION KEY						
n								(LSB)

[1] The Persistent Reservations Generation (PRGENERATION) value is a 32-bit counter in the target that shall be incremented every time a PERSISTENT RESERVE OUT command requests a Register, a Clear, a Preempt, or a Preempt and Abort service action. The counter shall not be incremented by a PERSISTENT RESERVE IN command, by a PERSISTENT RESERVE OUT command that performs a RESERVE or RELEASE service action, or by a PERSISTENT RESERVE OUT command that is not performed due to an error or reservation conflict. Regardless of the APTPL bit value, the GENERATION value shall be set to 0 as part of the POWER ON RESET process.

The GENERATION value allows the initiator examining the generation value to verify that the configuration of the initiators attached to a logical unit has not been modified by another initiator without the knowledge of the examining initiator.

[2] The ADDITIONAL LENGTH field contains a count of the number of bytes in the Reservation Key List. If the ALLOCATION LENGTH specified by the PERSISTENT RESERVE IN command is not sufficient to contain the entire parameter list, then only the bytes from 0 to the maximum allowed ALLOCATION LENGTH shall be sent to the initiator. The remaining bytes shall be truncated, although the ADDITIONAL LENGTH field shall still contain the actual number of bytes in the RESERVATION KEY LIST without consideration of any truncation resulting from an insufficient ALLOCATION LENGTH. This shall not be considered an error.

[3] The RESERVATION KEY LIST contains all the 8-byte reservation keys registered with the target through PERSISTENT RESERVE OUT, Reserve, Preempt, Preempt and Clear, or Register service actions. Each reservation key may be examined by the initiator and correlated with a set of initiators and SCSI ports by mechanisms outside the scope of this manual.

8.16.2 PERSISTENT RESERVE IN parameter data for READ Reservations

The format for the parameter data provided in response to a PERSISTENT RESERVE IN command with the READ Reservations service action is shown in Table 185.

Table 185. PERSISTENT RESERVE IN parameter data for READ Reservations

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	PRGENERATION [1]						
3								(LSB)
4	(MSB)	ADDITIONAL LENGTH (n–7) [2]						
7								(LSB)
8	(MSB)	RESERVATIONS DESCRIPTOR [3]						
n								(LSB)

[1] The Persistent Reservations Generation (PRGENERATION) value is a 32-bit counter in the target that shall be incremented every time a PERSISTENT RESERVE OUT command requests a Register, a Clear, a Preempt, or a Preempt and Clear operation. The counter shall not be incremented by a PERSISTENT RESERVE IN command, by a PERSISTENT RESERVE OUT command that performs a RESERVE or RELEASE service action, or by a PERSISTENT RESERVE OUT command that is not performed due to an error or reservation conflict. The Generation value shall be set to 0 as part of the power on reset process.

The GENERATION value allows the initiator examining the generation value to verify that the configuration of the initiators attached to a logical unit has not been modified by another initiator without the knowledge of the examining initiator.

[2] The ADDITIONAL LENGTH field contains a count of the number of bytes in of RESERVATIONS DESCRIPTOR. If the ALLOCATION LENGTH specified by the PERSISTENT RESERVE IN command is not sufficient to contain the entire parameter list, then only the bytes from 0 to the maximum allowed ALLOCATION LENGTH shall be sent to the initiator. The remaining bytes shall be truncated, although the ADDITIONAL LENGTH field shall still contain the actual number of bytes of RESERVATIONS DESCRIPTOR and shall not be affected by the truncation. This shall not be considered an error.

[3] The format of a single READ RESERVATIONS DESCRIPTOR is defined in Table 186. There shall be a RESERVATIONS DESCRIPTOR for the persistent reservation, if any, present in the logical unit and a RESERVATIONS DESCRIPTOR for each element, if any, having a persistent reservation.

Table 186. PERSISTENT RESERVE IN READ Reservations Descriptor

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	RESERVATION KEY [1]						
7								(LSB)
8	(MSB)	SCOPE-SPECIFIC ADDRESS [2]						
11								(LSB)
12	Reserved							
13	SCOPE [3]				TYPE [4]			
14	OBSOLETE [2]							
15								

[1] If a persistent reservation is present in the logical unit that does contain elements, there shall be a Reservation descriptor in the list of parameter data returned by the target in response to the PERSISTENT RESERVE IN command with a READ Reservation service action for the Logical Unit persistent reservation that is held, if any, and each element persistent reservation that may be held. The Reservation descriptor shall contain the Reservation Key under which the persistent reservation is held. The Type and Scope of the persistent reservation as present in the PERSISTENT RESERVE OUT command that created the persistent reservation shall be returned (see also notes [3] and [4]).

[2] If the SCOPE is an Element reservation, the SCOPE-SPECIFIC ADDRESS field shall contain the Element address, zero filled in the most significant bytes to fit the field. If the SCOPE is a Logical Unit reservation, the SCOPE-SPECIFIC ADDRESS bytes shall be set to zero. The OBSOLETE field in Bytes 14 and 15 was used for extent reservations as described in an earlier standard.

[3] The value in the SCOPE field shall indicate whether a persistent reservation applies to an entire logical unit, to a portion of the logical unit defined as an extent, or to an element. The values in the SCOPE field are defined in Table 187.

Table 187: Persistent Reservation Scope Codes

Code	Name	Description
00h	LU	Persistent reservation applies to the full logical unit. A SCOPE field value of LU_SCOPE shall indicate that the persistent reservation applies to the entire logical unit. The LU_SCOPE scope shall be implemented by all targets that implement PERSISTENT RESERVE OUT.
01h	Extent	Obsolete.
02h	Element	Persistent reservation applies to the specified element. A SCOPE field value of ELEMENT_SCOPE shall indicate that the persistent reservation applies to the element of the logical unit defined by the SCOPE-SPECIFIC ADDRESS field in the PERSISTENT RESERVE OUT parameter list. An element is defined by the SMC-2 standard. The ELEMENT_SCOPE scope is optional for all targets that implement PERSISTENT RESERVE OUT.
03h-1Fh	Reserved	Reserved.

[4] The value in the TYPE field shall specify the characteristics of the persistent reservation being established within the element or within the logical unit. Table 188 defines the characteristics of the five different type values. For each persistent reservation type, Table 188 lists code value and describes the required target support. In Table 188, the description of required target support is divided into three paragraphs. The first paragraph defines the required handling for read operations. The second paragraph defines the required handling for write operations.

Table 188: Persistent Reservation Type Codes

Code	Name	Description
00h	READ Shared	Obsolete.
01h	Write Exclusive	**Reads Shared**: Any initiator on any I_T nexus may initiate tasks that request transfers from the storage medium or cache of the logical unit to the I_T nexus. **Writes Exclusive**: Any task from any I_T nexus other than the I_T nexus holding the persistent reservation that requests a transfer from the I_T nexus to the storage medium or cache of the logical unit shall be terminated with RESERVATION CONFLICT status. **Persistent Reservation Holder:** The I_T nexus that delivered the PERSISTENT RESERVE OUT command with RESERVE, PREEMPT, or PREEMPT AND ABORT service action as identified by its registered reservation key.
02h	READ Exclusive	Obsolete.

Table 188: Persistent Reservation Type Codes (continued)

Code	Name	Description
03h	Exclusive Access	**Reads Exclusive:** Any task from any I_T nexus other than the I_T nexus holding the persistent reservation that requests a transfer from the storage medium or cache of the logical unit to the I_T nexus shall be terminated with RESERVATION CONFLICT status. **Writes Exclusive:** Any task from any I_T nexus other than the I_T nexus holding the persistent reservation that requests a transfer from the I_T nexus to the storage medium or cache of the logical unit shall be terminated with RESERVATION CONFLICT status. **Persistent Reservation Holder:** The I_T nexus that delivered the PERSISTENT RESERVE OUT command with RESERVE, PREEMPT, or PREEMPT AND ABORT service action as identified by its registered reservation key.
04h	Shared Access	Obsolete.
05h	Write Exclusive, Registrants Only	**Reads Shared**: Any initiator on any I_T nexus may initiate tasks that request transfers from the storage medium or cache of the logical unit to the I_T nexus. **Writes Exclusive:** A task that requests a transfer to the storage medium or cache of the logical unit from a initiator port that is not associated with a registered I_T nexus shall be terminated with RESERVATION CONFLICT status. **Persistent Reservation Holder:** The I_T nexus that delivered the PERSISTENT RESERVE OUT command with RESERVE, PREEMPT, or PREEMPT AND ABORT service action as identified by its registered reservation key.
06h	Exclusive Access, Registrants Only	**Reads Exclusive:** A task that requests a transfer from the storage medium or cache of the logical unit to a initiator port that is not associated with a registered I_T nexus shall be terminated with RESERVATION CONFLICT status. **Writes Exclusive:** A task that requests a transfer to the storage medium or cache of the logical unit from an initiator port that is not currently registered with the target shall be terminated with RESERVATION CONFLICT status. **Persistent Reservation Holder:** The I_T nexus that delivered the PERSISTENT RESERVE OUT command with RESERVE, PREEMPT, or PREEMPT AND ABORT service action as identified by its registered reservation key.
07h	Write Exclusive, All Registrants	**Reads Shared:** Any initiator on any I_T nexus may initiate tasks that request transfers from the storage medium or cache of the logical unit to the I_T nexus. **Writes Exclusive:** A task that requests a transfer to the storage medium or cache of the logical unit from a initiator port that is not associated with a registered I_T nexus shall be terminated with RESERVATION CONFLICT status. **Persistent Reservation Holder:** Any registered I_T nexus as identified by a zero reservation key value.
08h	Exclusive Access – All Registrants	**Reads Exclusive:** A task that requests a transfer from the storage medium or cache of the logical unit to a initiator port that is not associated with a registered I_T nexus shall be terminated with RESERVATION CONFLICT status. **Writes Exclusive:** A task that requests a transfer to the storage medium or cache of the logical unit from a initiator port that is not associated with a registered I_T nexus shall be terminated with RESERVATION CONFLICT status. **Persistent Reservation Holder:** Any registered I_T nexus as identified by a zero reservation key value.
09h-1Fh	Reserved	

8.16.3 PERSISTENT RESERVE IN parameter data for REPORT CAPABILITIES

The format for the parameter data provided in response to a PERSISTENT RESERVE IN command with the REPORT CAPABILITIES service action is shown in Table 103.

Table 189. PERSISTENT RESERVE IN parameter data for REPORT CAPABILITIES

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	LENGTH (0008h) [1]						
1								(LSB)
2	Reserved				SIP_C [2]	ATP_C [3]	ES_C [4]	PTPL_C [5]
3	Reserved							PTPL_A [6]
4	Reserved							
7								

[1] The LENGTH field indicates the length in bytes of the parameter data. If the ALLOCATION LENGTH field in the CDB is too small to transfer all of the parameter data, the length shall not be adjusted to reflect the truncation.

[2] An SIP_C (Specify Initiator Ports Capable) bit of one indicates that the target supports the SPEC_I_PT bit in the PERSISTENT RESERVE OUT command parameter data (see 6.12.3). An SIP_C bit of zero indicates that the target does not support the SPEC_I_PT bit in the PERSISTENT RESERVE OUT command parameter data.

[3] An ATP_C (All Target Ports Capable) bit of one indicates that the target supports the ALL_TG_PT bit in the PERSISTENT RESERVE OUT command parameter data. An ATP_C bit of zero indicates that the target does not support the ALL_TG_PT bit in the PERSISTENT RESERVE OUT command parameter data.

[4] An ES_C (Element Scope Capable) bit of one indicates that the target supports a SCOPE value of ELEMENT_SCOPE (see 6.11.4.2) in PERSISTENT RESERVE OUT commands (see 6.12). An ES_C bit of zero indicates that the target does not support a SCOPE value of ELEMENT_SCOPE in PERSISTENT RESERVE OUT commands.

[5] An PTPL_C (Persist Through Power Loss Capable) bit of one indicates that the target supports the persist through power loss capability (see 5.5.2.2) for persistent reservations and the APTPL bit in the PERSISTENT RESERVE OUT command parameter data. An PTPL_C bit of zero indicates that the target does not support the persist through power loss capability.

[6] A PTPL_A (Persist Through Power Loss Activated) bit of one indicates that persist through power loss capability (see 5.5.2.2) is activated because the most recent successfully completed PERSISTENT RESERVE OUT command with REGISTER or REGISTER AND IGNORE EXISTING KEY service action had the APTPL bit set to one in the parameter data. A PTPL_A bit of zero indicates that the persist through power loss capability is not activated.

8.17 PERSISTENT RESERVE OUT command (5Fh)

The PERSISTENT RESERVE OUT command (see Table 190) is used to request service actions that reserve a logical unit or element for the exclusive or shared use of a particular I_T nexus. The command uses other service actions to manage and remove such persistent reservations.

I_T nexuses performing PERSISTENT RESERVE OUT service actions are identified by a registered reservation key provided by the initiator. An initiator may use the PERSISTENT RESERVE IN command to obtain the reservation key, if any, for the I_T nexus holding a persistent reservation and may use the PERSISTENT RESERVE OUT command to preempt that persistent reservation.

If a PERSISTENT RESERVE OUT command is attempted, but there are insufficient target resources to complete the operation, the target shall return a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INSUFFICIENT REGISTRATION RESOURCES.

Table 190. PERSISTENT RESERVE OUT command (5Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (5Fh)							
1	Reserved			SERVICE ACTION [3]				
2	SCOPE [1]				TYPE [1]			
3	Reserved							
4	Reserved							
5	Reserved							
6	Reserved							
7	(MSB)	PARAMETER LIST LENGTH [2]						
8								(LSB)
9	CONTROL [4]							

[1] The PERSISTENT RESERVE OUT command contains fields that specify a persistent reservation Service Action, the intended scope of the persistent reservation, and the restrictions caused by the persistent reservation. The TYPE and SCOPE fields are defined in notes [3] and [4] of Table 186. If a SCOPE field specifies a scope that is not implemented, the target shall return a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and additional sense data shall be set to INVALID FIELD IN CDB.

Fields contained in the PERSISTENT RESERVE OUT parameter list specify the reservation keys and extent information required to perform a particular persistent reservation service action.

[2] The parameter list shall be 24 bytes in length and the PARAMETER LIST LENGTH field shall contain 24 (18h). If the PARAMETER LIST LENGTH is not 24, the target shall return a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense data shall be set to PARAMETER LIST LENGTH Error.

[3] When processing the PERSISTENT RESERVE OUT service actions, the target shall increment the generation value as specified in 8.16.1.

The PERSISTENT RESERVE OUT command Service Action codes are briefly described in Table 191.

Table 191. PERSISTENT RESERVE OUT Service Action codes

Code	Name	Description
00h	Register	The PERSISTENT RESERVE OUT command executing a Register service action registers a reservation key with a target. See T10/1416-D, Section 5.5.2.4, for details.
01h	Reserve	The PERSISTENT RESERVE OUT command performing a Reserve service action creates a persistent reservation having a specified scope and type. The scope and type of a persistent reservation are defined in notes [3] and [4] of Table 186. See T10/1416-D, Section 5.5.2.5, for details.
02h	Release	The PERSISTENT RESERVE OUT command performing a Release service action removes an active persistent reservation held by the same initiator. See T10/1416-D, Section 5.2.7.2, for details.
03h	Clear	The PERSISTENT RESERVE OUT command that successfully performs a Clear service action shall remove all persistent reservations for all initiators. See T10/1416-D, Section 5.5.2.7.6, for details.
04h	Preempt	Preempts persistent reservations from another initiator. See T10/1416-D, Section 5.5.2.7.4, for details.
05h	Preempt and Abort	Preempts persistent reservations from another initiator and aborts the task set for the preempted initiator. See T10/1416-D, Section 5.5.2.7.5, for details.
06h	Register and Ignore Existing Key	Register a reservation key with the target (see 5.5.2.4) or unregister a reservation key. See T10/1416-D, Section 5.5.2.7.3, for details.
07h-1Fh	Reserved	

[4] See Table 1, “CONTROL field,” in Section 7.2.1.

8.17.1 PERSISTENT RESERVE OUT parameter list

The parameter list required to perform the PERSISTENT RESERVE OUT command are defined in Table 192. All fields shall be sent on all PERSISTENT RESERVE OUT commands, even if the field is not required for the specified Service action and Scope values.

Table 192. PERSISTENT RESERVE OUT parameter list

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	RESERVATION KEY [1]						
7								(LSB)
8	(MSB)	SERVICE ACTION RESERVATION KEY [2]						
15								(LSB)
16	(MSB)	SCOPE-SPECIFIC ADDRESS						
19								(LSB)
20	Reserved				SPEC_I_PT	ALL_TG_PT	Reserved	APTPL [6]
21	Reserved							
22	Obsolete							
23								
24	ADDITIONAL PARAMETER DATA							
n								

[1] The RESERVATION KEY field contains an 8-byte token provided by the initiator to the target to identify the initiator that is the source of the PERSISTENT RESERVE OUT command. The target shall verify that the RESERVATION KEY field in a PERSISTENT RESERVE OUT command matches the registered reservation key for the initiator from which the command was received. If a PERSISTENT RESERVE OUT command specifies a RESERVATION KEY field other than the reservation key registered for the initiator, the target shall return a RESERVATION CONFLICT status. The reservation key of the initiator shall be valid for all Service action and SCOPE values. The obsolete field in Bytes 22 and 23 was used for extent reservations.

[2] The SERVICE ACTION RESERVATION KEY field contains information needed for three service actions; the Register, Preempt, and Preempt and Clear service actions. For the Register service action, the SERVICE ACTION RESERVATION KEY field contains the new reservation key to be registered. For the Preempt and Preempt and Clear service actions, the SERVICE ACTION RESERVATION KEY field contains the reservation key of the persistent reservations that are being preempted. For the Preempt and Preempt and Clear service actions, failure of the SERVICE ACTION RESERVATION KEY to match any registered reservation keys shall result in the target returning a RESERVATION CONFLICT status. The Service Action Reservation key is ignored for all service actions except those described in this paragraph.

[3] If the Scope is an Element reservation, the SCOPE-SPECIFIC ADDRESS field shall contain the Element address, zero filled in the most significant bytes to fit the field. If the Service action is Register or Clear or if the Scope is a Logical Unit reservation, the SCOPE-SPECIFIC ADDRESS field shall be set to zero.

[4] If the SPEC_I_PT (Specify Initiator Ports) bit is set to zero, the target shall ignore the additional parameter data and shall apply the registration only to the I_T nexus that sent the PERSISTENT RESERVE OUT command. If the SPEC_I_PT bit is set to one for the REGISTER or REGISTER AND IGNORE EXISTING KEY service actions, the additional parameter data shall include a list of transport IDs (see Table 107) and the target shall apply the registration to the I_T nexus for each initiator port specified by a TransportID. If a registration fails for any initiator port (e.g., if the SCSI target device does not have enough resources available to hold the registration information), none of the other registrations shall be made.

Table 193. PERSISTENT RESERVE OUT specify initiator ports additional parameter data

Bit Byte	7	6	5	4	3	2	1	0
24	TRANSPORTID PARAMETER DATA LENGTH (n - 27) [a]							
27								
	TRANSPORTIDs LIST							
28	FIRST TRANSPORTID							
:	:							
	LAST TRANSPORTID							
n								

[a] The TRANSPORTID PARAMETER DATA LENGTH field specifies the number of bytes of TransportIDs that follow.

The command shall be terminated with a CHECK CONDITION status and the sense key set to ILLEGAL REQUEST:

- If the value in the parameter list length field in the CDB does not include all of the additional parameter list bytes specified by the TRANSPORTID PARAMETER DATA LENGTH field; or
- If the value in the TRANSPORTID PARAMETER DATA LENGTH field results in the truncation of a TransportID.

The format of a TransportID is specified in SPC-3, Section 7.5.4.

[5] The ALL_TG_PT (All Target Ports) bit is valid only for the REGISTER and REGISTER AND IGNORE EXISTING KEY service actions, and shall be ignored for all other service actions. Support for the ALL_TG_PT bit is optional. If the target receives a REGISTER or REGISTER AND IGNORE EXISTING KEY service action with the ALL_TG_PT bit set to one, it shall create the specified registration on all target ports in the target device (i.e., as if the same registration request had been received individually through each target port). If the target receives a REGISTER or REGISTER AND IGNORE EXISTING KEY service action with the ALL_TG_PT bit set to zero, it shall apply the registration only to the target port through which the PERSISTENT RESERVE OUT command was received.

[6] The Activate Persist Through Power Loss (APTPL) bit shall be valid only for the Register service action. In all other cases, the APTPL shall be ignored. Support for an APTPL bit equal to one is optional. If a target that does not support the APTPL bit value of one receives that value in a Register service action, the target shall return a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and additional sense data shall be set to INVALID FIELD IN PARAMETER LIST.

If the last valid APTPL bit value received by the target is zero, the loss of power in the target shall release all persistent reservations and remove all reservation keys (see Table 191, register description). If the last valid APTPL bit value received by the target is one, the logical unit shall retain all persistent reservations and all reservation keys for all initiators even if power is lost and later returned. The most recently received valid APTPL value from any initiator shall govern logical unit's behavior in the event of power loss.

Table 194 summarizes which fields are set by the initiator and interpreted by the target for each SERVICE ACTION and SCOPE value. Two PERSISTENT RESERVE OUT parameters are not summarized in Table 194; Reservation key and APTPL, since they are specified in Table 192.

Table 194. PERSISTENT RESERVE OUT Service Actions and valid parameters

Service Action	**Allowed Scope**	**Parameters (part 1 of 2)**			
		Type	**Reservatio n Key**	**Service Action Reservation Key**	**Scope-Specific Address**
Register	ignored	ignored	valid	valid	ignored
Register and Ignore Existing Key	ignored	ignored	ignored	valid	ignored
Reserve	LU Element	valid	valid	ignored	ignored valid
Release	LU Element	valid	valid	ignored	ignored valid
Clear	ignored	ignored	valid	ignored	ignored
Preempt	LU Element	valid	valid	valid	ignored valid
Preempt and clear	LU Element	valid	valid	valid	ignored valid

Service Action	**Allowed Scope**	**Parameters (part 2 of 2)**		
		Type	**Reservation Key**	**Service Action Reservation Key**
Register	ignored	valid	valid	valid
Register and Ignore Existing Key	ignored	valid	valid	valid
Reserve	All	ignored	ignored	ignored
Release	All	ignored	ignored	ignored
Clear	ignored	ignored	ignored	ignored
Preempt	All	ignored	ignored	ignored
Preempt and clear	All	ignored	ignored	ignored

8.18 PREFETCH (10) command (34h)

The PREFETCH (10) command (Table 195) requests that the drive read and transfer the specified logical blocks to the drive's cache memory. No data shall be transferred to the initiator.

Table 195. PREFETCH (10) command (34h)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (34h)							
1	RESERVED [1]			Reserved			IMMED [2]	RELADR [3]
2	(MSB)							
3	LOGICAL BLOCK ADDRESS [4]							
4								
5								(LSB)
6	Reserved							
7	(MSB)	TRANSFER LENGTH [5]						
8								(LSB)
9	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] An Immediate (IMMED) bit of one indicates that status shall be returned as soon as the command descriptor block has been validated. An Immed bit of zero indicates that status shall be returned after the operation is complete.

[3] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[4] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [3]).

[5] The PREFETCH LENGTH field specifies the number of contiguous logical blocks of data that shall be transferred to the drive's cache memory. A PREFETCH LENGTH value of zero indicates that the contiguous logical blocks up to and including the last logical block of the logical unit shall be transferred to the drive's cache memory. Any other value indicates the number of logical blocks that shall be transferred. The drive may elect to not transfer logical blocks that already are contained in the cache memory.

If the Immed bit is zero and the specified logical blocks were successfully transferred to the cache memory, the drive shall return CONDITION MET status. If the Link bit (see Table 1, “CONTROL field,” in Section 7.2.1) is one, the drive shall return INTERMEDIATE–CONDITION MET status.

If IMMED is one, and the unlocked cache memory has sufficient capacity to accept all of the specified logical blocks, the drive shall return CONDITION MET status. If the LINK bit (see Table 1, “CONTROL field,” in Section 7.2.1) is one, the drive shall return INTERMEDIATE–CONDITION MET status.

If IMMED is one, and the unlocked cache memory does not have sufficient capacity to accept all of the specified logical blocks, the drive shall return GOOD status. The drive shall transfer to cache memory as many logical blocks as will fit. If the LINK bit (see Table 1, “CONTROL field,” in Section 7.2.1) is one, the drive shall return INTERMEDIATE status.

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

8.19 PREFETCH (16) command (90H)

The PREFETCH (16) command (Table 196) requests that the drive read and transfer the specified logical blocks to the drive's cache memory. No data shall be transferred to the initiator.

Table 196. PREFETCH (16) command (90h)

Bit Byte	7	6	5	4	3	2	1	0
0	OPERATION CODE (90h)							
1	RESERVED [1]						IMMED [2]	RELADR [3]
2	(MSB)	LOGICL BLOCK ADDRESS [4]						
3								
4								
5								
6								
7								
8								
9								(LSB)
10	(MSB)	PREFETCH LENGTH [5]						
11								
12								
13								(LSB)
14	Reserved							
15	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] An Immediate (IMMED) bit of one indicates that status shall be returned as soon as the command descriptor block has been validated. An Immed bit of zero indicates that status shall be returned after the operation is complete.

[3] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[4] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [3]).

[5] The PREFETCH LENGTH field specifies the number of contiguous logical blocks of data that shall be transferred to the drive's cache memory. A PREFETCH LENGTH value of zero indicates that the contiguous logical blocks up to and including the last logical block of the logical unit shall be transferred to the drive's cache memory. Any other value indicates the number of logical blocks that shall be transferred. The drive may elect to not transfer logical blocks that already are contained in the cache memory.

If the Immed bit is zero and the specified logical blocks were successfully transferred to the cache memory, the drive shall return CONDITION MET status. If the Link bit (see Table 1, “CONTROL field,” in Section 7.2.1) is one, the drive shall return INTERMEDIATE–CONDITION MET status.

If IMMED is one, and the unlocked cache memory has sufficient capacity to accept all of the specified logical blocks, the drive shall return CONDITION MET status. If the LINK bit (see Table 1, “CONTROL field,” in Section 7.2.1) is one, the drive shall return INTERMEDIATE–CONDITION MET status.

If IMMED is one, and the unlocked cache memory does not have sufficient capacity to accept all of the specified logical blocks, the drive shall return GOOD status. The drive shall transfer to cache memory as many logical blocks as will fit. If the LINK bit (see Table 1, “CONTROL field,” in Section 7.2.1) is one, the drive shall return INTERMEDIATE status.

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

8.20 PREVENT/ALLOW MEDIUM REMOVAL command (1Eh)

Not implemented by the drive. If received, the drive terminates with CHECK CONDITION status and sets an Illegal Request Sense key.

8.21 READ (6) command (08h)

The READ (6) command (Table 197) requests that the drive transfer data to the initiator. The most recent data value written, or to be written if cached, in the addressed logical block shall be returned.

Table 197. READ (6) command (08h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	0	0
1	RESERVED [1]			LOGICAL BLOCK ADDRESS (MSB) [2]				
2	LOGICAL BLOCK ADDRESS							
3	LOGICAL BLOCK ADDRESS (LSB)							
4	TRANSFER LENGTH [3]							
5	CONTROL [4]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] The LOGICAL BLOCK ADDRESS specifies that logical block at which the Read operation shall begin.

[3] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data to be transferred. A TRANSFER LENGTH of zero indicates that 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks that shall be transferred. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

IMPLEMENTORS NOTE: The READ (6) command is limited to directly addressing logical blocks up to a capacity of 2 Gigabytes, for logical block sizes of 512 bytes, this command has been maintained as mandatory since some system initialization routines require that the READ (6) command be used.

[4] See Table 1, “CONTROL field,” in Section 7.2.1.

The data value most recently written in the addressed logical block shall be returned.

Read data transfers with the initiator do not begin until at least one full sector of data is available in the drive data buffer. For multiple sector reads, the transfer of data continues until the number of blocks specified in byte 4 of the CDB has been read and transferred or until an unrecoverable error is detected.

Data transfer could stop if the option to stop on recovered error is selected.

For systems that support disconnection, the drive disconnects when a valid READ command is received. The drive reconnects depending on the value of the Buffer Full Ratio Set in Page 2 of the MODE SELECT Data (see Section 8.11). After data transfer has been initiated with a SCSI initiator port, the drive does not disconnect unless an internal error recovery procedure is required or the data transfer to a SCSI initiator port is interrupted for more than 1 millisecond.

The initiator must accept all data presented to the initiator after sending this command until the drive sends Completion Status during a STATUS phase.

Note. The drive may disconnect and reconnect while executing this command and the initiator may prematurely terminate this command by creating the Reset condition or by sending an ABORT, ABORT TAG, CLEAR QUEUE, or BUS DEVICE RESET message.

Sense Data is valid after this command is executed and Completion Status is sent. If the ADDRESS VALUE bit in the Sense Data is true (1), the Sense Data LOGICAL BLOCK ADDRESS (Information bytes) points to the last Logical Block accessed by the drive. If the ADDRESS VALID bit in the Sense Data is false (0), the Sense Data LOGICAL BLOCK ADDRESS bytes are not valid.

The drive contains a large buffer and implements an optional “prefetch” and segmented cache function whereby the requested read data is read into the buffer, plus an additional amount, depending on the cache control parameters. See section “PREFETCH and multisegmented cache control” in the individual drive’s Product Manual, Volume 1, for more information on this.

This command is terminated with a RESERVATION CONFLICT status and no data is read if any reservation access conflict (see Section 8.39) exists.

If any of the following conditions occur, this command shall be terminated with a CHECK CONDITION status, and if extended sense is implemented, the Sense Key shall be set as indicated in the following table. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	Sense Key
INVALID LOGICAL BLOCK ADDRESS	ILLEGAL REQUEST (see note below)
TARGET RESET SINCE LAST COMMAND FROM THIS INITIATOR	UNIT ATTENTION
UNRECOVERABLE READ ERROR	MEDIUM ERROR
RECOVERED READ ERROR	RECOVERED ERROR

Note. The extended sense information bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.22 READ (10) command (28h)

The READ (10) command (formerly called the READ EXTENDED command) requests that the target transfer data to the initiator. This command is implemented with the drive specific parameters shown in Table 198.

This command operates the same as the READ (6) command (see Section 8.21) except that in the CDB for this command a four byte LOGICAL BLOCK ADDRESS and a two byte Transfer Length may be specified.

The data value most recently written in the addressed logical block is returned to the Host.

This command shall be terminated with a RESERVATION CONFLICT status if any reservation access conflict (see Section 8.39) exists, and no data shall be read.

Table 198. READ (10) command (28h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	0	1	0	0	0
1	0	0	0	DPO [2]	FUA [3]	0	0	RELADR [4]
	RESERVED [1]							
2	LOGICAL BLOCK ADDRESS (MSB) [5]							
3	LOGICAL BLOCK ADDRESS [5]							
4	LOGICAL BLOCK ADDRESS [5]							
5	LOGICAL BLOCK ADDRESS (LSB) [5]							
6	0	0	0	0	0	0	0	0
7	TRANSFER LENGTH (MSB) [6]							
8	TRANSFER LENGTH (LSB) [6]							
9	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the drive shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the host knows the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the host expects that logical blocks accessed by this command are likely to be accessed again in the near future.

[3] A Force Unit Access (FUA) bit of one indicates that the target shall access the media in performing the command prior to returning GOOD status. READ commands shall access the specified logical blocks from the media (i.e., the data is not directly retrieved from the cache). In the case where the cache contains a more recent version of a logical block than the media, the logical block shall first be written to the media.

An FUA bit of zero indicates that the target may satisfy the command by accessing the cache memory. For READ operations, any logical blocks that are contained in the cache memory may be transferred to the initiator directly from the cache memory.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A TRANSFER LENGTH of zero indicates that no logical blocks shall be transferred. This condition shall not be considered an error. Any other value indicates the number of logical blocks that shall be transferred.

IMPLEMENTORS NOTE: The READ (10) command is limited to directly addressing logical blocks up to a capacity of 2 Terabytes with logical block sizes of 512 bytes.

[7] See Table 1, "CONTROL field," in Section 7.2.1.

If any of the following conditions occur, this command shall return a CHECK CONDITION status and the Sense Key shall be set as indicated. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	**Sense Key**
INVALID LOGICAL BLOCK ADDRESS	ILLEGAL REQUEST (see note)
TARGET RESET SINCE LAST COMMAND FROM THIS INITIATOR	UNIT ATTENTION
UNRECOVERED READ ERROR	MEDIUM ERROR
RECOVERABLE READ ERROR	RECOVERED ERROR

Note. The extended sense information bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.23 READ (12) command (A8h)

The READ (12) command requests that the target transfer data to the initiator.

Table 199. READ (12) command (A8h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	0	1	0	0	0
1	0	0	0	DPO [2]	FUA [3]	0	0	RELADR [4]
	RESERVED [1]							
2	LOGICAL BLOCK ADDRESS (MSB) [5]							
3	LOGICAL BLOCK ADDRESS							
4	LOGICAL BLOCK ADDRESS							
5	LOGICAL BLOCK ADDRESS (LSB)							
6	TRANSFER LENGTH (MSB) [6]							
7	TRANSFER LENGTH							
8	TRANSFER LENGTH							
9	TRANSFER LENGTH (LSB)							
10	Reserved							
11	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the drive shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the host knows the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the host expects that logical blocks accessed by this command are likely to be accessed again in the near future.

[3] A Force Unit Access (FUA) bit of one indicates that the target shall access the media in performing the command prior to returning GOOD status. READ commands shall access the specified logical blocks from the media (i.e., the data is not directly retrieved from the cache). In the case where the cache contains a more recent version of a logical block than the media, the logical block shall first be written to the media.

An FUA bit of zero indicates that the target may satisfy the command by accessing the cache memory. For READ operations, any logical blocks that are contained in the cache memory may be transferred to the initiator directly from the cache memory.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command.

This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS field specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A TRANSFER LENGTH of zero indicates that no logical blocks shall be transferred. This condition shall not be considered an error. Any other value indicates the number of logical blocks that shall be transferred.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

If any of the following conditions occur, this command shall return a CHECK CONDITION status and the Sense Key shall be set as indicated. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	**Sense Key**
INVALID LOGICAL BLOCK ADDRESS	ILLEGAL REQUEST (see note)
TARGET RESET SINCE LAST COMMAND FROM THIS INITIATOR	UNIT ATTENTION
UNRECOVERED READ ERROR	MEDIUM ERROR
RECOVERABLE READ ERROR	RECOVERED ERROR

Note. The extended sense information bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.24 READ (16) command (88h)

The READ (16) command requests that the target transfer data to the initiator.

Table 200. READ (16) command (88h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	0	0
1	0	0	0	DPO [2]	FUA [3]	0	0	RELADR [4]
	RESERVED [1]							
2	LOGICAL BLOCK ADDRESS (MSB) [5]							
3 - 8	LOGICAL BLOCK ADDRESS [5]							
9	LOGICAL BLOCK ADDRESS (LSB) [5]							
10	TRANSFER LENGTH (MSB) [6]							
11 - 12	TRANSFER LENGTH [6]							
13	TRANSFER LENGTH (LSB) [6]							
14	Reserved							
15	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the drive shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the host knows the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the host expects that logical blocks accessed by this command are likely to be accessed again in the near future.

[3] A Force Unit Access (FUA) bit of one indicates that the target shall access the media in performing the command prior to returning Good status. READ commands shall access the specified logical blocks from the media (i.e., the data is not directly retrieved from the cache). In the case where the cache contains a more recent version of a logical block than the media, the logical block shall first be written to the media.

An FUA bit of zero indicates that the target may satisfy the command by accessing the cache memory. For READ operations, any logical blocks that are contained in the cache memory may be transferred to the initiator directly from the cache memory.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A TRANSFER LENGTH of zero indicates that no logical blocks shall be transferred. This condition shall not be considered an error. Any other value indicates the number of logical blocks that shall be transferred.

IMPLEMENTORS NOTE: The READ (16) command may address logical blocks of capacities greater than 2 Terabytes with logical block sizes of 512 bytes.

[7] See Table 1, "CONTROL field," in Section 7.2.1.

If any of the following conditions occur, this command shall return a CHECK CONDITION status and the Sense Key shall be set as indicated. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	**Sense Key**
INVALID LOGICAL BLOCK ADDRESS	ILLEGAL REQUEST (see note)
TARGET RESET SINCE LAST COMAND FROM THIS INITIATOR	UNIT ATTENTION
UNRECOVERED READ ERROR	MEDIUM ERROR
RECOVERABLE READ ERROR	RECOVERED ERROR

Note. The extended sense information bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.25 READ BUFFER command (3Ch)

The READ BUFFER command is used in conjunction with the WRITE BUFFER command as a diagnostic function for testing drive memory and the SCSI bus integrity. This command shall not alter the medium. Command format is shown in Table 201.

Table 201. READ BUFFER command (3Ch)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	1	1	1	0	0
1	0	0	0	0	MODE [2] [6]			
	RESERVED [1]			Reserved				
2	BUFFER ID [5]							
3	(MSB)							
4	BUFFER OFFSET [5]							
5								(LSB)
6	(MSB)							
7	ALLOCATION LENGTH [3]							
8								(LSB)
9	CONTROL [4]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] READ BUFFER Mode field. The interpretation of data bits 0-3 is given in the following table and referenced sections.

DB3	DB2	DB1	DB0	Mode definition
0	0	0	0	Read combined descriptor header and data (Section 8.25.1)
0	0	1	0	Read data (Section 8.25.2)
0	0	1	1	Read descriptor (Section 8.25.3)
0	1	0	0	Reserved
0	1	0	1	Reserved
0	1	1	0	Reserved
0	1	1	1	Reserved
1	0	1	0	Read data from echo buffer (Section 8.25.4) [6]
1	0	1	1	Echo buffer descriptor mode (Section 8.25.5) [6]
1	0	0	0	Reserved
1	0	0	1	Reserved
1	1	0	0	Reserved
:	:	:	:	Reserved
1	1	1	1	Reserved

[3] Drive specific size also depends on whether cache is enabled as shown in Table 161 (RCD = 0) or not (RCD = 1). See individual drive's Product Manual, Volume 1, under list of SCSI commands supported.

[4] See Table 1, "CONTROL field," in Section 7.2.1.

[5] Not implemented by drives supported by this manual. Must be zero.

[6] The Read data from echo buffer mode and Echo buffer descriptor mode do not apply to drives marketed before the current release of this manual. Drives marketed after this release may or may not support these two modes. See individual drive's Product Manual, Volume 1, for declaration of support.

8.25.1 Read Combined Descriptor Header and Data mode (0000b)

In this mode, a four byte READ BUFFER header followed by the data bytes are returned to the initiator during the DATA IN phase. The BUFFER ID and the BUFFER OFFSET fields are not used by drives supported by this manual, and must be zero (see Table 202).

Table 202. READ BUFFER header (0000b)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	(MSB)							
2	BUFFER CAPACITY [1]							
3								(LSB)

[1] The BUFFER CAPACITY field specifies the total number of data bytes that are available in the drive's data buffer (see sections 8.64.1 and 8.64.3). This number is not reduced to reflect the ALLOCATION LENGTH nor is it reduced to reflect the actual number of bytes written using the WRITE BUFFER command. Following the READ BUFFER header, the drive shall transfer data from its data buffer. The drive terminates the DATA IN phase when ALLOCATION LENGTH bytes of header plus data have been transferred or when all available header and buffer data have been transferred to the initiator, whichever is less.

8.25.2 READ DATA mode (0010b)

In this mode, the DATA IN phase contains buffer data only with no header. The BUFFER ID and BUFFER OFFSET fields are not used.

8.25.3 READ BUFFER Descriptor mode (0011b)

In this mode, a maximum of four bytes of READ BUFFER descriptor information is returned. If there is no buffer associated with the specified BUFFER ID, the target shall return all zeros in the READ BUFFER descriptor. The BUFFER OFFSET field is reserved in this mode. The ALLOCATION LENGTH should be set to four or greater. The target shall transfer the lesser of the allocation length or four bytes of READ BUFFER descriptor. The READ BUFFER descriptor is defined as shown in Table 203.

IMPLEMENTORS NOTE: In a multi-tasking system, a buffer may be altered between the WRITE BUFFER and READ BUFFER commands by another task. Buffer testing applications may wish to insure that only a single task is active. Use of reservations (to all logical units on the device) may also be helpful in avoiding buffer alteration between these two commands.

Table 203. READ BUFFER Descriptor mode (0011b)

Bit Byte	7	6	5	4	3	2	1	0
0	OFFSET BOUNDARY [1]							
1	(MSB)							
2	BUFFER CAPACITY [2]							
3								(LSB)

[1] The OFFSET BOUNDARY field returns the boundary alignment with the selected buffer for subsequent WRITE BUFFER and READ BUFFER commands. The value contained in the OFFSET BOUNDARY field shall be interpreted as a power of two.

The value contained in the buffer field of subsequent WRITE BUFFER and READ BUFFER commands should be a multiple of $2^{\text{offset boundary}}$ as shown in Table 204.

Table 204. Buffer Offset Boundary

Offset Boundary	$2^{\text{Offset Boundary}}$	Buffer Offsets
0	$2^0 = 1$	Byte boundaries
1	$2^1 = 2$	Even-byte boundaries
2	$2^2 = 4$	Four-byte boundaries
3	$2^3 = 8$	Eight-byte boundaries
4	$2^4 = 16$	16-byte boundaries
:	:	:
FFh	Not applicable	0 is the only supported buffer offset

[2] The BUFFER CAPACITY field specifies the total number of data bytes that are available in the drive’s data buffer (see sections 8.64.1 and 8.64.3). This number is not reduced to reflect the ALLOCATION LENGTH nor is it reduced to reflect the actual number of bytes written using the WRITE BUFFER command. Following the READ BUFFER header, the drive shall transfer data from its data buffer. The drive terminates the DATA IN phase when ALLOCATION LENGTH bytes of header plus data have been transferred or when all available header and buffer data have been transferred to the initiator, whichever is less.

8.25.4 Read Data from Echo Buffer mode (1010b)

In this mode, the target transfers data to the initiator from the echo buffer. The echo buffer shall transfer the same data as when the WRITE BUFFER command with the mode field set to echo buffer was issued. The BUFFER ID and BUFFER OFFSET fields are ignored in this mode.

Note. A WRITE BUFFER command with the mode field set to echo buffer should be sent prior to the READ BUFFER command; otherwise the READ BUFFER command may terminate with CHECK CONDITION: ILLEGAL REQUEST or may return indeterminate data.

The READ BUFFER command shall return the same number of bytes of data as received in the prior WRITE BUFFER command.

8.25.5 Echo Buffer Descriptor mode (1011b)

In this mode, a maximum of four bytes of READ BUFFER descriptor information is returned. The target shall return the descriptor information for the echo buffer. If there is no echo buffer implemented, the target shall return all zeros in the READ BUFFER descriptor. The BUFFER OFFSET field is reserved in this mode. The ALLOCATION LENGTH should be set to four or greater. The target shall transfer the lesser of the ALLOCATION LENGTH or four bytes of READ BUFFER descriptor. The READ BUFFER descriptor is defined as shown in Table 205.

Table 205. Echo Buffer Descriptor mode (1011b)

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							EBOS [1]
1	Reserved							
2	Reserved			(MSB)				
3	BUFFER CAPACITY [2]							(LSB)

[1] If the echo buffer is implemented then the echo buffer descriptor shall be implemented. An echo buffer overwritten supported (EBOS) bit of one indicates either:

[a] The target returns the ECHO BUFFER OVERWRITTEN additional sense code if the data being read from the echo buffer is not the data previously written by the same initiator, or

[b] The target ensures echo buffer data from each initiator is the same as that previously written by the same initiator.

An EBOS bit of zero specifies that the echo buffer may be overwritten by other initiators or intervening commands.

[2] The BUFFER CAPACITY field shall return the size of the echo buffer in bytes aligned to a four-byte boundary.

8.26 READ CAPACITY (10) command (25h)

The READ CAPACITY (10) command (Table 206) provides a means for the initiator to request the capacity of the drive information.

Table 206. READ CAPACITY (10) command (25h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	0	0	1	0	1
1	RESERVED [1]			Reserved				RELADR [2]
2	(MSB)							
:	RETURNED LOGICAL BLOCK ADDRESS [2]							
5								(LSB)
6 - 7	Reserved							
8	Reserved							PMI [4]
9	CONTROL [5]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[3] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin, if RELADR bit is zero (see note [2]).

[4] A Partial Medium Indicator (PMI) bit of zero indicates the information returned in the READ CAPACITY data shall be the LOGICAL BLOCK ADDRESS and BLOCK LENGTH (in bytes) of the last logical block of the logical unit. (This value is the same for all drives of the same model number, sector size, and sparing scheme. It is the same regardless of the number of defective blocks the drive has.) The LOGICAL BLOCK ADDRESS in the Command Descriptor Block shall be set to zero for this option.

A PMI bit of one indicates the information returned shall be the LOGICAL BLOCK ADDRESS and BLOCK LENGTH (in bytes) of the last LOGICAL BLOCK ADDRESS after which a substantial delay (defined as approximately 1 millisecond for the typical drive) in data transfer is encountered. This returned LOGICAL BLOCK ADDRESS shall be greater than or equal to the LOGICAL BLOCK ADDRESS specified in the Command Descriptor Block. This reported LOGICAL BLOCK ADDRESS is the last block prior to a cylinder boundary.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

In response to the READ CAPACITY command, the drive returns 8 bytes of READ CAPACITY (16) Data to the host. The contents of the 8 bytes are listed in Table 207.

Table 207. READ CAPACITY Data)

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	RETURNED LOGICAL BLOCK ADDRESS						
3								(LSB)
4	(MSB)	BLOCK LENGTH IN BYTES						
7								(LSB)

8.27 READ CAPACITY (16) command (9Eh)

The READ CAPACITY (16) command (Table 208) provides a means for the initiator to request information regarding the capacity of the block device. This command is implemented as a service action of the SERVICE ACTION IN operation code.

Table 208. READ CAPACITY (16) command (9Eh)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	0	1	1	1	1	0
1	RESERVED [1]			SERVICE ACTION (10h) [2]				
2	(MSB)	LOGICAL BLOCK ADDRESS [3]						
:								
9								(LSB)
10	(MSB)	ALLOCATION LENGTH [4]						
:								
13								(LSB)
14	RESERVED						RELADR [5]	PMI [6]
15	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] 10h is the Service Action Code for the READ CAPACITY (16) command.

[3] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall begin.

[4] ALLOCATION LENGTH specifies the number of bytes the initiator has allocated for the returned defect data. An ALLOCATION LENGTH of zero indicates that no READ DEFECT DATA shall be transferred. Any other value indicates the maximum number of bytes to be transferred. The drive shall terminate the DATA IN phase when the ALLOCATION LENGTH bytes have been transferred or when all available defect data has been transferred to the initiator, whichever is less.

[5] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[6] A Partial Medium Indicator (PMI) bit of zero indicates the information returned in the READ CAPACITY data shall be the LOGICAL BLOCK ADDRESS and BLOCK LENGTH (in bytes) of the last logical block of the logical unit. (This value is the same for all drives of the same model number, sector size, and sparing scheme. It is the same regardless of the number of defective blocks the drive has.) The LOGICAL BLOCK ADDRESS in the Command Descriptor Block shall be set to zero for this option.

A PMI bit of one indicates the information returned shall be the LOGICAL BLOCK ADDRESS and BLOCK LENGTH (in bytes) of the last LOGICAL BLOCK ADDRESS after which a substantial delay (defined as approximately 1 millisecond for the typical drive) in data transfer is encountered. This returned LOGICAL

BLOCK ADDRESS shall be greater than or equal to the LOGICAL BLOCK ADDRESS specified in the Command Descriptor Block. This reported LOGICAL BLOCK ADDRESS is the last block prior to a cylinder boundary.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

In response to the READ CAPACITY (16) command, the drive returns 12 bytes of READ CAPACITY (16) Data to the host. The contents of the 12 bytes are listed in Table 209.

Table 209. READ CAPACITY (16) Data

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	RETURNED LOGICAL BLOCK ADDRESS						
7								(LSB)
8	(MSB)	BLOCK LENGTH IN BYTES						
11								(LSB)

8.28 READ DEFECT DATA (10) command (37h)

The READ DEFECT DATA (10) command requests that the target transfer the medium defect data to the initiator. If the drive is unable to access any medium defect data, it terminates the command with CHECK CONDITION status. The sense key is set to either MEDIUM ERROR (if a medium error occurred) or NO SENSE (if the list does not exist) and the additional sense code is set to DEFECT LIST NOT FOUND.

This command is implemented with the drive specific parameters listed in Table 210.

Table 210. READ DEFECT DATA (10) command (37h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	1	0	1	1	1
1	RESERVED [1]			Reserved				
2	0	0	0	P	G	1	0	0 or 1
				[2]		DEFECT LIST FORMAT [3]		
3 - 6	0	0	0	0	0	0	0	0
7	(MSB)	ALLOCATION LENGTH [4]						
8								(LSB)
9	CONTROL [5]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] The drive interprets the P and G bits (bits 4 and 3 of byte 2 of the CDB) as follows:

Bit P	Bit G	
0	0	Return Defect List header only. Target returns only the Defect List header.
0	1	Return the growth “G” list only. This list reflects the growth or “G” list as defined in Section 8.5.
1	0	Return the manufacturer’s original ETF list only. This list reflects the manufacturer’s original ETF list. These defects may or may not have been reallocated, depending on the last format command received (the last format may or may not have requested the P list flaws be reallocated during the format function).
1	1	Return all lists. The returned list contains all of the requested drive’s defect lists (i.e., P, G, C, and D) regardless of whether these lists have been reallocated by the drive.

[3] The DEFECT LIST FORMAT field indicates the defect data format preferred by the initiator. The meaning is the same as the DEFECT LIST FORMAT field in the FORMAT UNIT command (indicated in Table 65).

The DEFECT LIST FORMAT bits (bits 2, 1, 0 in the CDB) should be: 1 0 0, respectively, to signify a defect list in the Bytes from Index format, or 1 0 1, respectively, to signify a defect list in the Physical Sector format. If neither of these two, the drive responds with the defect list in the drives default format (physical sector) and creates the CHECK CONDITION status with RECOVERED ERROR SENSE KEY (1h) and additional sense error code (1C) at the end of the READ DEFECT DATA transfer.

[4] ALLOCATION LENGTH specifies the number of bytes the initiator has allocated for the returned defect data. An ALLOCATION LENGTH of zero indicates that no READ DEFECT DATA shall be transferred. Any other value indicates the maximum number of bytes to be transferred. The drive shall terminate the DATA

IN phase when the ALLOCATION LENGTH bytes have been transferred or when all available defect data has been transferred to the initiator, whichever is less.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

This command is intended to be used only with the FORMAT UNIT command (Section 8.5). The initiator should not interpret or act upon this list except to resend this list as defect data in a FORMAT UNIT command. It is not possible to relate actual physical locations to logical block addresses given in connection with other commands.

The format Defect Data Header and Defect Data Descriptor bytes returned are shown in Table 211.

The first 4 bytes returned are the Defect List Header. The P bit, G bit, and Defect List Format fields indicate the defect format actually returned by the drive. The definitions are the same as for byte 2 of the READ DEFECT DATA Command Descriptor Block (Table 210).

The Defect List Length specifies the total length (in bytes) of all the defect descriptors available from the drive. If the ALLOCATION LENGTH of the CDB is too small to transfer all of the defect descriptors, the Defect List Length is not adjusted to reflect the truncation. The drive does not create the CHECK CONDITION status. The Defect Descriptors are not required to be in ascending order.

Table 211. Defect List Header Description

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved			P	G	1	0	0 or 1
				[1]		DEFECT LIST FORMAT [2]		
2	DEFECT LIST LENGTH (MSB) (Value is 8 times the number of defects) [3]							
3	DEFECT LIST LENGTH (LSB) [3]							
4-n	DEFECT DESCRIPTOR BYTES [3]							

[1] The drive interprets the P and G bits (bits 4 and 3 of byte 1 of the CDB) as follows:

Bit P	Bit G	
0	0	Return Defect List header only. Target returns only the Defect List header.
0	1	Return the growth “G” list only. This list reflects the growth or “G” list as defined in Section 8.5.
1	0	Return the manufacturer’s original ETF list only. This list reflects the manufacturer’s original ETF list. These defects may or may not have been reallocated, depending on the last Format command received (the last format may or may not have requested the P list flaws be reallocated during the format function).
1	1	Return all lists. The returned list contains all of the requested drive’s defect lists (i.e., P, G, C, and D) regardless of whether these lists have been reallocated by the drive.

[2] 1 0 0 defines a list in the Bytes from Index format.
1 0 1 defines a list in the Physical Sector format.

[3] If the P and G bits are 0 0, the Defect List Length will be zero and no Defect Descriptor Bytes are sent to the initiator. See tables 74 and 75 for format of Defect Descriptor Bytes.

8.29 READ DEFECT DATA (12) command (B7h)

The READ DEFECT DATA (12) command (see Table 212) requests that the target transfer the medium defect data to the initiator.

If the logical unit is reserved, a reservation conflict shall occur when a READ DEFECT DATA (12) command is received from a SCSI initiator port other then the one holding a logical unit reservation. The command shall be rejected with RESERVATION CONFLICT status if the reservation conflict is due to a logical unit reservation. READ DEFECT DATA (12) commands with a reservation conflict shall be terminated with RESERVATION CONFLICT status. The READ DEFECT DATA (12) command shall not be evaluated for extent reservation conflicts (e.g., extent reservations do not conflict with the READ DEFECT DATA (12) command).

Table 212. READ DEFECT DATA (12) command (B7h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	1	0	1	1	1
1	RESERVED [1]			P LIST[2]	G LIST [2]	DEFECT LIST FORMAT [3]		
2	Reserved							
:								
5								
6	(MSB)							
:	ALLOCATION LENGTH [4]							
9								(LSB)
10	Reserved							
11	CONTROL [5]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] The drive interprets the P and G bits (bits 4 and 3 of byte 2 of the CDB) as follows:

Bit P	Bit G	
0	0	Return Defect List header only. Target returns only the Defect List header.
0	1	Return the growth “G” list only. This list reflects the growth or “G” list as defined in Section 8.5.
1	0	Return the manufacturer’s original ETF list only. This list reflects the manufacturer’s original ETF list. These defects may or may not have been reallocated, depending on the last Format command received (the last format may or may not have requested the P list flaws be reallocated during the format function).
1	1	Return all lists. The returned list contains all of the requested drive’s defect lists (i.e., P, G, C, and D) regardless of whether these lists have been reallocated by the drive.

[3] The DEFECT LIST FORMAT field indicates the defect data format preferred by the initiator. The meaning is the same as the DEFECT LIST FORMAT field in the FORMAT UNIT command (indicated in Table 65).

The DEFECT LIST FORMAT bits (bits 2, 1, 0 in the CDB) should be: 1 0 0, respectively, to signify a defect list in the Bytes from Index format, or 1 0 1, respectively, to signify a defect list in the Physical Sector format. If neither of these two, the drive responds with the defect list in the drives default format (physical sector) and creates the CHECK CONDITION status with Recovered Error Sense Key (1h) and additional sense error code (1C) at the end of the READ DEFECT DATA transfer.

[4] ALLOCATION LENGTH specifies the number of bytes the initiator has allocated for the returned defect data. An ALLOCATION LENGTH of zero indicates that no READ DEFECT DATA shall be transferred. Any other value indicates the maximum number of bytes to be transferred. The drive shall terminate the DATA IN phase when the ALLOCATION LENGTH bytes have been transferred or when all available defect data has been transferred to the initiator, whichever is less.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

The READ DEFECT DATA (12) list header (see Table 213) contains an eight byte header, followed by zero or more defect descriptors.

Table 213. READ DEFECT DATA (12) list header

Bit Byte	7	6	5	4	3	2	1	0
0	Reserved							
1	Reserved			P LIST [1]	G LIST [1]	DEFECT LIST FORMAT [2]		
2 - 3	Reserved							
4	(MSB)							
:	DEFECT LIST LENGTH [3]							
7								(LSB)
Defect Descriptors [3]								
8	DEFECT DESCRIPTOR BYTES [3]							
n								

[1] The drive interprets the P and G bits (bits 4 and 3 of byte 1 of the CDB) as follows:

Bit P	Bit G	
0	0	Return Defect List header only. Target returns only the Defect List header.
0	1	Return the growth “G” list only. This list reflects the growth or “G” list as defined in Section 8.5.
1	0	Return the manufacturer’s original ETF list only. This list reflects the manufacturer’s original ETF list. These defects may or may not have been reallocated, depending on the last format command received (the last format may or may not have requested the P list flaws be reallocated during the format function).
1	1	Return all lists. The returned list contains all of the requested drive’s defect lists (i.e., P, G, C, and D) regardless of whether these lists have been reallocated by the drive.

[2] 1 0 0 defines a list in the Bytes from Index format.
1 0 1 defines a list in the Physical Sector format.

[3] If the P and G bits are 0 0, the DEFECT LIST LENGTH will be zero and no Defect Descriptor Bytes are sent to the initiator. See tables 74 and 75 for format of Defect Descriptor Bytes.

8.30 READ ELEMENT STATUS command (B4h)

Not described in this manual. If received, a CHECK CONDITION status is sent.

8.31 READ LONG command (3Eh)

The READ LONG command (Table 214) requests that the target transfers data to the initiator. The data passed during the READ LONG command shall include the data bytes, followed by the ECC bytes of the single logical block addressed by the command. The most recent data written, or to be written, in the addressed logical block shall be returned. READ LONG is independent of the Read-Write Error Recovery mode page but does allow retries.

Table 214. READ LONG command (3Eh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	1	1	1	1	0
1	RESERVED [1]			Reserved			CORRECT [2]	RELADR [3]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [4]							
5								(LSB)
6	Reserved							
7	(MSB)	BYTE TRANSFER LENGTH [5]						
8								(LSB)
9	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A CORRECT bit of zero requests that a logical block be read without any ECC correction made by the target. A CORRECT bit of one requests that the data be corrected by ECC, if necessary.

Other error recovery procedures separate from ECC are applied in accordance with the parameter settings given in the MODE SENSE Error Recovery page (01h). See individual drive's Product Manual, Volume 1, for a detailed discussion of the drive error recovery procedures.

If the DCR bit of the READ-WRITE Error Recovery page is equal to one and the CORRECT bit equals one, the READ LONG command shall be terminated with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST with an additional sense code of Invalid Field in CDB.

[3] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first RELADR logical block of the range of logical blocks to be operated on by this command.

[4] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall occur. The most recent data written in the addressed logical block shall be returned.

[5] The BYTE TRANSFER LENGTH field species the number of bytes of data that should be transferred. If a non-zero BYTE TRANSFER LENGTH does not match the available data length, the target shall terminate the command with CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST with the additional sense code set to INVALID FIELD IN CDB. The VALID and ILI bits (see SPC-3) shall be set to one and the INFORMATION field shall be set to the difference (residue) of the requested length

minus the actual length in bytes. Negative values shall be indicated by two's complement notation.

A BYTE TRANSFER LENGTH of zero indicates that no bytes shall be transferred and shall not be considered an error. The BYTE TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

8.32 REASSIGN BLOCKS command (07h)

The REASSIGN BLOCKS command (Table 215) requests the target to reassign the defective logical blocks to an area on the logical unit reserved for this purpose.

After sending the REASSIGN BLOCKS command, the initiator transfers a defect list that contains the logical block addresses to be reassigned. The drive reassigns the physical medium used for each LOGICAL BLOCK ADDRESS in the list. The data contained in the logical blocks specified in the defect list is not preserved, but the data in all other logical blocks on the medium is preserved. It is recommended that the initiator recover the data from the logical blocks to be reassigned before issuing this command. After completion of this command, the initiator can write the recovered data to the same logical block addresses.

The effect of specifying a logical block to be reassigned that has previously been reassigned is to reassign the block again. Thus, over the life of the medium, a logical block can be assigned to multiple physical addresses (until no more spare locations remain on the medium).

This command should be used by a SCSI initiator port to immediately reallocate any block (sector) which requires the drive to recover data by data correction via ECC if the automatic reallocation feature of the drive is not enabled (see MODE SELECT command, Section 8.11).

For systems which support disconnection, the drive disconnects while executing this command.

Table 215. REASSIGN BLOCKS command (07h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	1	1
1	RESERVED [1]			Reserved			LONGLBA [2]	LONGLIST [3]
2	Reserved							
3								
4								
5	CONTROL [4]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Long LBA (LONGLBA) bit of zero requests that four byte defect descriptors be returned in the REASSIGN BLOCKS defect list. A LONGLBA bit of one requests that eight byte defect descriptors be returned in the REASSIGN BLOCKS defect list.

[3] A LONGLIST bit of zero indicates that the defect list header follows the short format in Table 69. A LONGLIST bit of one indicates that the defect list header follows the long format in Table 69.

[4] See Table 1, “CONTROL field,” in Section 7.2.1.

8.32.1 REASSIGN BLOCKS defect list

The REASSIGN BLOCKS defect list (Table 216) contains a four byte header followed by one or more Defect Descriptors. The length of each Defect Descriptor is four bytes.

Table 216. REASSIGN BLOCKS defect list

Bit Byte	7	6	5	4	3	2	1	0
Defect List Length Header								
0	DEFECT LIST LENGTH [1]							
3								
Defect Descriptor(s) [2]								
0	DEFECT LOGICAL BLOCK ADDRESS 0							
3 or 7								
	. . .							
n- (3 or 7)	DEFECT LOGICAL BLOCK ADDRESS X							
n								

[1] The DEFECT LIST LENGTH specifies the total length in bytes of the Defect Descriptors that follow. The DEFECT LIST LENGTH is equal to four times (if the LONGLBA bit is set to zero) or eight times (if the LONGLBA bit is set to one) the number of Defect Descriptors and does not include the defect list header length.

[2] The Defect Descriptor contains the DEFECT LOGICAL BLOCK ADDRESS. The logical block address is a four-byte field (if the LONGLBA bit is set to zero) or an eight-byte field (if the LONGLBA bit is set to one). The defect descriptors shall be in ascending order.

If the logical unit has insufficient capacity to reassign all of the defective logical blocks, the command shall terminate with a CHECK CONDITION status and the Sense Key shall be set to Hardware Error and the additional sense code set to NO DEFECT SPARE LOCATION AVAILABLE. The LOGICAL BLOCK ADDRESS of the first logical block not reassigned shall be returned in the information bytes of the sense data. If information about the first defect descriptor not reassigned is not available, or if all the defects have been reassigned, this field shall be set to FFFFFFFFh.

If the REASSIGN BLOCKS command failed due to an unexpected unrecoverable read error that would cause the loss of data in a block not specified in the defect list, the LOGICAL BLOCK ADDRESS of the unrecoverable block shall be returned in the information field of the sense data and the valid bit shall be set to one.

IMPLEMENTORS NOTE: If the REASSIGN BLOCKS command returns CHECK CONDITION status and the sense data command-specific information field contains a valid LOGICAL BLOCK ADDRESS, the initiator should remove all defect descriptors from the defect list prior to the one returned in the command-specific information field. If the sense key is MEDIUM ERROR and the VALID bit is one (the information field contains the valid block address), the initiator should insert that new defective LOGICAL BLOCK ADDRESS into the defect list and reissue the REASSIGN BLOCKS command with the new defect list. Otherwise, the initiator should perform any corrective action indicated by the sense data and then reissue the REASSIGN BLOCKS command with the new defect list.

8.33 RECEIVE DIAGNOSTIC RESULTS command (1Ch)

The RECEIVE DIAGNOSTIC RESULTS command requests analysis data be sent to the initiator after completion of a SEND DIAGNOSTIC command. The drive supports the optional Page format, wherein the initiator sends additional pages after a SEND DIAGNOSTIC command. These additional pages have a page code that specifies to the drive the format of the data to be returned after it receives a RECEIVE DIAGNOSTIC RESULTS command.

If no data in the optional Page format was requested by the SEND DIAGNOSTICs command (0 in bit 4 of Table 237), the data returned to the initiator is in the format shown in Table 218.

If the SEND DIAGNOSTICs command requested either page 00h or page 40h (the only two optional pages supported by the drive), data returned is in the format shown in Table 219 or 220, respectively.

All FRU and error code definitions are unique to this product and intended for factory/field maintenance personnel.

Table 217. RECEIVE DIAGNOSTIC RESULTS command (1Ch)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	0	0
1	RESERVED [1]			Reserved				PCV [2]
2	PAGE CODE [3]							
3	ALLOCATION LENGTH in Bytes (MSB) [4]							
4	ALLOCATION LENGTH in Bytes (LSB) [4]							
5	CONTROL [5]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A page code valid (PCV) bit of zero indicates that the most recent SEND DIAGNOSTIC command shall define the data returned by this command. Optionally, a PCV bit of one indicates that the contents of the PAGE CODE field shall define the data returned by this command.

[3] The PAGE CODE field identifies which diagnostic page is requested as a result of a RECEIVE DIAGNOSTIC RESULTS command with the PCV bit equal to one, or returned as a result of a RECEIVE DIAGNOSTIC RESULTS parameter data.

[4] The ALLOCATION LENGTH shall specify the number of bytes the initiator has allocated for returned data. An ALLOCATION LENGTH of zero indicates that no data shall be transferred. Any other value indicates the maximum number of bytes that shall be transferred. The drive terminates the DATA IN phase when ALLOCATION LENGTH bytes have been transferred or when all available data has been transferred to the initiator, whichever is less.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

Table 218. Diagnostic Data Bytes

Code	Byte	Description
00h	0	ADDITIONAL LENGTH (MSB) [1]
28h	1	ADDITIONAL LENGTH (LSB) [1]
XXh	2	FRU CODE (most probable) [2]
XXh	3	FRU CODE [2]
XXh	4	FRU CODE [2]
XXh	5	FRU CODE (least probable) [2]
XXh	6	ERROR CODE (MSB) [3]
V.U.	7	ERROR CODE (LSB) [4]
V.U.	8 thru n	ADDITIONAL VENDOR UNIQUE FAULT INFORMATION [5]

[1] ADDITIONAL LENGTH. This two byte value indicates the number of additional bytes included in the diagnostic data list. For example, if no product unique byte (byte 7) is available, this value would be 0006h. A value of 0000h means that there are no additional bytes.

[2] A FIELD REPLACEABLE UNIT (FRU) Code is a byte that identifies an assembly that may have failed. The codes will be listed in probability order, with the most probable assembly listed first and the least probable listed last. A code of 00h indicates there is no FRU information and a code of 01h indicates the entire unit should be replaced. Seagate drives return 00h in these bytes.

[3] The ERROR CODE is a two byte value that provides information designating which part of a diagnostic operation has failed. The byte 7 error code is vendor unique and defined in note [4]. Usually, Seagate drives support only some subset of the list given in note [4].

[4] Vendor Unique codes supported by Seagate devices.

Diagnostic Fault Codes

01h	Formatter Diagnostic Error
02h	Microprocessor RAM Diagnostic Error
04h	No Drive Ready
08h	No Sector or Index Detected
09h	Fatal Hardware Error While Doing Drive Diagnostics
0Ch	No Drive Command Complete
10h	Unable to Set Drive Sector Size
14h	Unable to Clear Drive Attention
18h	Unable to Start Spindle Motor
20h	Unable to Recal Drive
30h	Unable to Send Write Current Data to Drive
34h	Unable to Issue Drive SEEK Command
40h	Unable to Read User Table From Drive
41h	Ran Out of Sectors While Doing Drive Diagnostics
42h	Unable to Read Reallocation Table
43h	Unable to Read ETF Log
45h	Firmware Read from Disc or Sent by Host has an Invalid Checksum
60h	Thermal Calibration Failure
70h	Microprocessor Internal Timer Error
80h	Buffer Controller Diagnostic Error
81h	Buffer RAM Diagnostic Error
C1h	Data Miscompare While Doing Drive Diagnostics

[5] ADDITIONAL VENDOR UNIQUE CODES (not available).

8.33.1 Supported Diagnostic Pages

If the SEND DIAGNOSTICs command requests the Supported Diagnostics Pages list (PF bit = 1), the drive returns data in the format shown in Table 219 after receiving the RECEIVE DIAGNOSTIC RESULTS command. It lists all of the diagnostics pages supported by the drive.

Table 219. Supported Diagnostic Pages

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (00h)							
1	Reserved							
2	(MSB)	PAGE LENGTH (n-3) [1]						
3								(LSB)
4 : n	SUPPORTED PAGE LIST [2]							

[1] The PAGE LENGTH field specifies the length in bytes of the following SUPPORTED PAGE LIST.

[2] The SUPPORTED PAGE LIST field shall contain a list of all diagnostic page codes implemented by the drive in ascending order beginning with page code 00h. The drive presently supports only pages 00h (Table 219) and 40h (Table 220).

8.33.2 Translate Address page (40h)

The Translate Address page allows the initiator to translate a LOGICAL BLOCK ADDRESS into a physical sector address or a physical sector address to a LOGICAL BLOCK ADDRESS. The address to be translated is passed to the target during the DATA OUT phase associated with the SEND DIAGNOSTIC command and the results are returned to the initiator during the DATA IN phase following the RECEIVE DIAGNOSTIC RESULTS command. The translated address is returned in the Translate Address page—Receive Diagnostic (Table 220).

Table 220. Translate Address page—Receive Diagnostic (40h)

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (40h) [1]							
1	Reserved							
2	(MSB)	PAGE LENGTH (000A) or (0002) [2]						
3								(LSB)
4	Reserved				SUPPLIED FORMAT [3]			
5	RAREA [4]	ALTSEC [5]	ALTTK [6]	Reserved	TRANSLATED FORMAT [7]			
6 : 13	TRANSLATED ADDRESS (if available) [8]							

[1] The Translate Address page contains a four byte page header which specifies the page code and length followed by two bytes which describe the translated address followed by the translated address.

[2] The PAGE LENGTH field contains the number of parameter bytes which follow.

[3] The SUPPLIED FORMAT field contains the value from the SEND DIAGNOSTIC command supplied format field (see Table 240).

[4] A Reserved Area (RAREA) bit of one indicates that all or part of the translated address falls within a reserved area of the medium (e.g., speed tolerance gap, alternate logical block, vendor reserved area, etc.). If the entire translated address falls within a reserved area the target may not return a translated address. An RAREA bit of zero indicates that no part of the translated address falls within a reserved area of the medium.

Table 221. Address Field Logical Block Address Format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)							
:	LOGICAL BLOCK ADDRESS							
3								(LSB)
4	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Table 222. Address Field Physical Sector Address Format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)							
1	CYLINDER NUMBER							
2								(LSB)
3	HEAD NUMBER							
4	SECTOR NUMBER							
5								
6								
7								

[5] An Alternate Sector (ALTSEC) bit of one indicates that the translated address is physically located in an alternate sector of the medium. If the drive cannot determine if all or part of the translated address is located in an alternate sector it shall set this bit to zero. An ALTSEC bit of zero indicates that no part of the translated address is located in an alternate sector of the medium or that the drive is unable to determine this information.

[6] An Alternate Track (ALTTRK) bit of one indicates that part or all of the translated address is located on an alternate track of the medium or the drive cannot determine if all or part of the translated address is located on an alternate track. An ALTTRK bit of zero indicates that no part of the translated address is located on an alternate track of the medium.

[7] The TRANSLATED FORMAT field contains the value from the SEND DIAGNOSTIC command translate format field (see Table 240). The values are 000 (Logical block format) or 101 (Physical sector address format).

[8] The TRANSLATED ADDRESS field contains the address the target translated from the address supplied by the initiator in the SEND DIAGNOSTIC command. This field shall be in the format specified in the translate format field. The supported formats are shown in Tables 221 and 222.

8.34 RELEASE (6) command (17h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.35 RELEASE (10) command (57h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.36 REPORT DEVICE IDENTIFIER command (A3h)

The REPORT DEVICE IDENTIFIER command (see Table 223) requests that the target send device identification information to the initiator. This command is optional for all device types, except for SCC-2 devices, and devices that set the SCCS bit in their Standard INQUIRY data (see Section 8.6.1). Only the REPORT DEVICE IDENTIFIER operation code and service action concerns all SCSI devices.

Table 223. REPORT DEVICE IDENTIFIER command (A3h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	0	0	0	1	1
1	Reserved			SERVICE ACTION (05h) [1]				
2	Reserved							
3								
4	See SCC-2 [2]							
5								
6	(MSB)	ALLOCATION LENGTH [3]						
:								
9								(LSB)
10	Reserved						See SCC-2 [4]	Reserved
11	CONTROL [5]							

[1] SERVICE ACTION. As defined in the SCC-2 standard (SCSI Controller Command set T10/1225), the REPORT DEVICE IDENTIFIER command is the Report Peripheral Device/Component Device Identifier service action of the MAINTENANCE IN command (also Op. code A3h, but for initiators). Additional Maintenance In service actions (that apply to SCC-2 devices and devices that set the SCCS bit in their Standard INQUIRY data) are defined in SCC-2, Section 6.3.

[2] SCC-2 defines specific usages for bytes 4 and 5, and bit 1 in byte 10, however these fields are reserved for the REPORT DEVICE IDENTIFIER command defined by this manual.

[3] The ALLOCATION LENGTH field indicates how much space has been reserved for the returned parameter data (see Table 223). If the length is not sufficient to contain all the parameter data, the first portion of the data shall be returned. This shall not be considered an error. The actual length of the parameter data is available in the Identifier Length field in the parameter data. If the remainder of the parameter data is required, the initiator should send a new REPORT DEVICE IDENTIFIER command with an ALLOCATION LENGTH field large enough to contain all the data.

[4] The Report Device Identifier parameter list (see Table 224) contains a four-byte field that contains the length in bytes of the parameter list and the logical unit's identifier.

Table 224: Report Device Identifier parameter list

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	IDENTIFIER LENGTH (n – 4) [a]						
1								
2								
3								(LSB)
4	IDENTIFIER [b]							
n								

[a] The IDENTIFIER LENGTH field specifies the length in bytes of the Identifier field. If the ALLOCATION LENGTH field in the CDB is too small to transfer all of the identifier, the length shall not be adjusted to reflect the truncation. The identifier length shall initially equal zero, and shall be changed only by a successful SET DEVICE IDENTIFIER command (see Section 8.48).

[b] The IDENTIFIER field shall contain a vendor-specific value. The value reported shall be the last value written by a successful SET DEVICE IDENTIFIER command. The value of the IDENTIFIER shall be changed only by a SET DEVICE IDENTIFIER command. The IDENTIFIER value shall persist through resets, power cycles, media format operations, and media replacement. The target shall return the same IDENTIFIER to all initiators on all ports.

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

The execution of a REPORT DEVICE IDENTIFIER may require the enabling of a nonvolatile memory within the logical unit. If the nonvolatile memory is not ready, the target shall return CHECK CONDITION status, rather than wait for the device to become ready. The sense key shall be set to NOT READY and the additional sense data shall be set as described in the TEST UNIT READY command (see Section 8.53). This information should allow the initiator to determine the action required to cause the target to become ready.

8.37 REPORT LUNS command (A0h)

The REPORT LUNS command (see table 225) requests that the peripheral device logical unit inventory accessible to the initiator via the addressed target port be sent to the initiator. The logical unit inventory is a list that shall include the logical unit numbers of all logical units having a PERIPHERAL QUALIFIER value of 000b (see 8.6.1). Logical unit numbers for logical units with PERIPHERAL QUALIFIER values of 100b, 101b, 110b, or 111b may optionally be included in the logical unit inventory.

A SCSI device shall support a REPORT LUNS command that is addressed to logical unit zero. Support of the REPORT LUNS command by logical units other than logical unit zero is optional. Support of the REPORT LUNS command on devices having only a single logical unit with the logical unit number of zero is optional.

The REPORT LUNS command shall not be affected by reservations or persistent reservations.

Table 225. REPORT LUNS command (A0h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	0	0	0	0	0
1	Reserved							
2	SELECT REPORT [1]							
3	Reserved							
4								
5								
6	(MSB)							
:	ALLOCATION LENGTH [2]							
9								(LSB)
10	Reserved							
11	CONTROL [3]							

[1] The SELECT REPORT field is defined in Table 226.

Table 226. Select report code values

Value	Description
00h	The logical unit addresses reported shall be limited to the following addressing methods (see SAM-2): a) Logical unit addressing method, b) Peripheral device addressing method; and c) Flat space addressing method.
01h	The list of logical units shall only contain well known logical units, if any. If there are no well known logical units the LUN LIST LENGTH field shall be zero.
02h	All logical units accessible to the initiator via the addressed target port shall be reported.
03h - FFh	Reserved

[2] The ALLOCATION LENGTH shall be at least 16 bytes. If the ALLOCATION LENGTH is less than 16 bytes, the target shall return CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense data shall be set to INVALID FIELD IN CDB.

The ALLOCATION LENGTH is not sufficient to contain the LOGICAL UNIT NUMBER values for all configured logical units, the target shall report as many LOGICAL UNIT NUMBER values as will fit in the specified ALLOCATION LENGTH. This shall not be considered an error.

[3] See Table 1, “CONTROL field,” in Section 7.2.1.

The REPORT LUNS command shall return CHECK CONDITION status only when the target is unable to return the requested report of the logical unit inventory. If a REPORT LUNS command is received from an initiator with a pending unit attention condition (i.e., before the target reports CHECK CONDITION status), the target shall perform the REPORT LUNS command.

If the unit attention condition was established because of a change in the logical unit inventory, that unit attention condition shall be cleared for that initiator by the REPORT LUNS command. Unit attention conditions established for other reasons shall not be cleared by the REPORT LUNS command (see SAM-2).

The REPORT LUNS data should be returned even though the target is not ready for other commands. The default report of the logical unit inventory should be available without incurring any media access delays. The default report of the logical unit inventory shall contain at least LUN 0.

If the logical unit inventory changes for any reason, including completion of initialization, removal of a logical unit, or creation of a logical unit, the target shall generate a unit attention command for all initiators (see SAM-2). The target shall set the additional sense code to REPORTED LUNS DATA HAS CHANGED.

The execution of a REPORT LUNS command to any valid and installed logical unit shall clear the REPORTED LUNS DATA HAS CHANGED unit attention condition for all logical units of that target with respect to the requesting initiator. A valid and installed logical unit is one having a PERIPHERAL QUALIFIER of 000b in the standard INQUIRY data (see 8.6.1).

The target shall report the LOGICAL UNIT NUMBERs of configured logical units using the format shown in Table 227.

Table 227. REPORT LUNS parameter list format

Bit Byte	7	6	5	4	3	2	1	0
0	(MSB)	LUN LIST LENGTH (n-7) [1]						
3								(LSB)
4	(MSB)	Reserved						
7								(LSB)
	LUN List							
8	(MSB)	FIRST LUN						
15								(LSB)
	. . .							
n–7	(MSB)	LAST LUN						
n								(LSB)

[1] The LUN LIST LENGTH field shall contain the length in bytes of the LUN list that is available to be transferred. The LUN LIST LENGTH is the number of logical unit numbers reported multiplied by eight. If the allocation length in the command descriptor block is too small to transfer information about all configured logical units, the LUN LIST LENGTH value shall not be adjusted to reflect the truncation.

8.38 REQUEST SENSE command (03h)

The REQUEST SENSE command (Table 228) requests that the target transfer sense data to the initiator in the format shown in Table 229.

If any nonfatal error occurs during execution of REQUEST SENSE, the drive shall return sense data with GOOD status. Following a fatal error on a REQUEST SENSE command, sense data may be invalid.

Table 228. REQUEST SENSE command (03h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1	1
1	Reserved							DESC [1]
2	Reserved							
3								
4	ALLOCATION LENGTH [2]							
5	CONTROL [3]							

[1] The DESC indicates which sense data format shall be returned. If DESC is set to zero, fixed format sense data (see SPC-3, Section 4.5.3) shall be returned. If DESC is set to one and descriptor format sense data (see SPC-3, Section 4.5.2) is supported, descriptor format sense data shall be returned.

[2] The ALLOCATION LENGTH in byte four of the format shown specifies the number of bytes the initiator has allocated for returned sense data. The ALLOCATION LENGTH should always be at least 18 bytes for drive devices for the initiator to receive all of the drive sense data. Any other value indicates the maximum number of bytes that shall be transferred. The drive shall terminate the DATA IN phase when ALLOCATION LENGTH bytes have been transferred or when all available sense data have been transferred to the initiator, whichever is less. The drive always returns sense data in the Extended Sense Data Format.

[3] See Table 1, “CONTROL field,” in Section 7.2.1.

Sense data shall be available and cleared under the conditions defined in SAM-2. If the target has no other sense data available to return, it shall return a sense key of No Sense and an additional sense code of No Additional Sense Information.

If the target is in the STANDBY POWER condition or IDLE POWER condition when a REQUEST SENSE command is received and there is no ACA or CA condition, the target shall return a sense key of NO SENSE and an additional sense code of LOW POWER CONDITION ON. On completion of the command the logical unit shall return to the same power condition that was active before the REQUEST SENSE command was received. A REQUEST SENSE command shall not reset any active power condition timers.

The target shall return CHECK CONDITION status for a REQUEST SENSE command only to report exception conditions specific to the command itself. For example:

a. An invalid field value is detected in the command descriptor block;
b. An unrecovered parity error is detected by the service delivery subsystem; or
c. a SCSI target port malfunction that prevents return of the sense data.

If a recovered error occurs during the execution of the REQUEST SENSE command, the target shall return the sense data with GOOD status. If a target returns CHECK CONDITION status for a REQUEST SENSE command, the sense data may be invalid.

Note. The sense data appropriate to the selection of an invalid logical unit is defined in SAM-2.

Targets shall be capable of returning eighteen bytes of data in response to a REQUEST SENSE command. If the ALLOCATION LENGTH is 18 or greater, and a target returns less than 18 bytes of data, the initiator should assume that the bytes not transferred would have been zeros had the target returned those bytes. initiators may determine how much sense data has been returned by examining the ALLOCATION LENGTH field in the command descriptor block and the Additional Sense Length field in the sense data. targets shall not adjust the additional sense length to reflect truncation if the ALLOCATION LENGTH is less than the sense data available.

The sense data format for response codes 70h (current errors) and 71h (deferred errors) are defined in Table 229.

Table 229. Response codes 70h and 71h sense data format

Bit Byte	7	6	5	4	3	2	1	0
0	VALID [1]	RESPONSE CODE (70h or 71h) [2]						
1	Obsolete [3]							
2	FILEMARK [4]	EOM [5]	ILI [6]	Reserved	SENSE KEY [7]			
3	(MSB)	INFORMATION [8]						
6								(LSB)
7	ADDITIONAL SENSE LENGTH (n – 7) [9]							
8	(MSB)	COMMAND-SPECIFIC INFORMATION [10]						
11								(LSB)
12	ADDITIONAL SENSE CODE [11]							
13	ADDITIONAL SENSE CODE QUALIFIER [12]							
14	FIELD REPLACEABLE UNIT CODE [13]							
15	SKSV [14]	SENSE KEY SPECIFIC [15]						
17								
18	ADDITIONAL SENSE BYTES [16]							
n								

[1] A VALID bit of zero indicates that the INFORMATION field is not as defined in this manual. A VALID bit of one indicates the INFORMATION field contains valid information as defined in this manual. targets shall implement the VALID bit.

[2] RESPONSE CODE value 70h (current errors) is described in Section 8.38.2. targets shall implement RESPONSE CODE 70h. RESPONSE CODE 71h (deferred errors) is described in Section 8.38.3. Implementation of RESPONSE CODE 71h is optional. RESPONSE CODE 7Fh is for a vendor-specific sense data formats. RESPONSE CODE values 72h to 7Eh and 00h to 6Fh are reserved.

[3] Obsolete. Seagate disc drives do not currently implement any of the COPY commands.
The SEGMENT NUMBER field contains the number of the current segment descriptor if the REQUEST SENSE command is in response to a COPY, COMPARE, or COPY AND VERIFY command. Up to 256 segments are supported, beginning with segment zero.

[4] The Filemark bit is mandatory for sequential-access devices, and this bit is reserved for all other device types. A Filemark bit of one indicates that the current command has read a filemark or setmark. The Additional Sense Code field may be used to indicate whether a filemark or setmark was read. Reporting of setmarks is optional and indicated by the RSMK bit for sequential-access devices in the configuration parameters page. (See SSC.)

[5] The End-of-Medium (EOM) bit is mandatory for sequential-access and printer devices, and this bit is reserved for all other device types. An EOM bit of one indicates that an end-of-medium condition (end-of-partition, beginning-of-partition, out-of-paper, etc.) exists. For sequential-access devices, this bit indicates that the unit is at or past the early-warning if the direction was forward, or that the command was not completed because beginning-of-partition was encountered if the direction was reverse. (See SSC.)

[6] An Incorrect Length Indicator (ILI) bit of one usually indicates that the requested logical block length did not match the logical block length of the data on the medium.

[7] The SENSE KEY, ADDITIONAL SENSE CODE and ADDITIONAL SENSE CODE QUALIFIER fields provide a hierarchy of information. The intention of the hierarchy is to provide a top-down approach for an initiator to determine information relating to the error and exception conditions. The sense key provides generic categories in which error and exception conditions may be reported. initiators typically use sense keys for high level error recovery procedures.

Additional sense codes provide further detail describing the sense key. Additional sense code qualifiers add further detail to the additional sense code. The additional sense code and additional sense code qualifier may be used by initiators where sophisticated error recovery procedures require detailed information describing the error and exception conditions.

The Sense Key field is mandatory and indicates generic information describing an error or exception condition. The sense keys are defined in Section 8.38.2.

[8] The contents of the INFORMATION field is device-type or command specific and is defined within the appropriate standard for the device type or command of interest. targets shall implement the INFORMATION field. Unless specified otherwise, this field contains:

a. the unsigned LOGICAL BLOCK ADDRESS associated with the sense key, for direct-access devices (device type 0), write-once devices (device type 4), CD-ROM devices (device type 5), and optical memory devices (device type 7);

b. the difference (residue) of the requested length minus the actual length in either bytes or blocks, as deter-mined by the command, for sequential-access devices (device type 1), printer devices (device type 2), processor devices (device type 3) and some direct access device commands, except as defined for d) below. Negative values are indicated by two's complement notation;

c. the difference (residue) of the requested number of blocks minus the actual number of blocks copied or compared for the current segment descriptor of a COPY, COMPARE, COPY AND VERIFY, or Extended COPY command; or

d. for sequential-access devices operating in buffered modes 1h or 2h that detect an unrecoverable write error when unwritten data blocks, filemarks, or setmarks remain in the buffer, the value of the information field for all commands shall be:

   - the total number of data blocks, filemarks, and setmarks in the buffer if the device is in fixed block mode (block length field of the MODE SENSE block descriptor is non-zero and the fixed bit of the WRITE command is one); or
   - the number of bytes in the buffer, including filemarks and setmarks, if the device is in variable mode (the fixed bit of the WRITE command is zero).

For additional information, see SSC (SCSI Stream Device Command Set, T10/997D).

[9] The ADDITIONAL SENSE LENGTH field indicates the number of additional sense bytes to follow. If the ALLOCATION LENGTH of the command descriptor block is too small to transfer all of the additional sense bytes, the additional sense length is not adjusted to reflect the truncation.

[10] The COMMAND-SPECIFIC INFORMATION field contains information that depends on the command that was executed. Further meaning for this field is defined within the command description. The COMMAND-

SPECIFIC INFORMATION field is mandatory if the target supports any of the following commands: COPY, COMPARE, COPY AND VERIFY, Extended COPY and REASSIGN BLOCKS.

[11] The Additional Sense Code (ASC) indicates further information related to the error or exception condition reported in the sense key field. targets shall support the ASC field. Support of the additional sense codes not explicitly required by this manual is optional. A list of additional sense codes is in Section 8.38.3. If the target does not have further information related to the error or exception condition, the additional sense code is set to No Additional Sense Information.

[12] The Additional Sense Code Qualifier (ASCQ) indicates detailed information related to the additional sense code. The additional sense code qualifier is optional. If the error or exception condition is reportable by the device, the value returned shall be as specified in Section 8.38.3. If the target does not have detailed information related to the error or exception condition, the additional sense code qualifier is set to zero.

[13] Non-zero values in the FIELF REPLACEABLE UNIT CODE field are used to define a device-specific mechanism or unit that has failed. A value of zero in this field shall indicate that no specific mechanism or unit has been identified to have failed or that the data is not available. The FIELD REPLACEABLE UNIT CODE field is optional. The format of this information is not specified by this manual. Additional information about the field replaceable unit may be available in the ASCII information page (see Section 8.6.5), if supported by the target.

[14] The SENSE KEY SPECIFIC bytes are described in Section 8.38.1.

[15] The ADDITIONAL SENSE BYTES field may contain command specific data, peripheral device specific data, or vendor-specific data that further defines the nature of the CHECK CONDITION status.

[16] Bytes 18-n are not presently used.

8.38.1 Sense Key Specific field

The SENSE KEY SPECIFIC field as defined by this manual when the value of the Sense Key Specific Valid (SKSV) bit is one. The SKSV bit and SENSE KEY SPECIFIC fields are optional. The definition of this field is determined by the value of the SENSE KEY field. This field is reserved for sense keys not described below. An SKSV value of zero indicates that this field is not as defined by this manual.

Table 230. Field pointer bytes

Bits Bytes	7	6	5	4	3	2	1	0
15	SKSV [1]	C/D [2]	Reserved		BPV [3]	BIT POINTER [4]		
16	(MSB)	FIELD POINTER [5]						
17								(LSB)

[1] SKSV. If the sense key is ILLEGAL REQUEST and the SKSV bit is set to one, the SENSE KEY SPECIFIC field (Table 229) shall be as defined as shown in this table. The FIELD POINTER field indicates which illegal parameters in the command descriptor block or the data parameters are in error.

[2] A Command Data (C/D) bit of one indicates that the illegal parameter is in the command descriptor block. A C/D bit of zero indicates that the illegal parameter is in the data parameters sent by the initiator in the Data Out Buffer.

[3] A Bit Pointer Valid (BPV) bit of zero indicates that the value in the Bit Pointer field [4] is not valid. A BPV bit of one indicates that the BIT POINTER field specifies which bit of the byte designated by the FIELD POINTER field is in error.

[4] When a multiple-bit field is in error, the BIT POINTER field shall point to the most-significant (left-most) bit of the field.

[5] The FIELD POINTER field indicates which byte of the command descriptor block or of the parameter data was in error. Bytes are numbered starting from zero, as shown in the tables describing the commands and parameters.

When a multiple-byte field is in error, the field pointer shall point to the most significant (left-most) byte of the field. If several consecutive bytes are reserved, each shall be treated as a single-byte field.

Note. Bytes identified as being in error are not necessarily the place that has to be changed to correct the problem.

If the sense key is RECOVERED ERROR (1h), HARDWARE ERROR (4h), or MEDIUM ERROR (3h), and if the SKSV bit is one, the SENSE KEY SPECIFIC field is defined as shown in Table 231. These fields identify the actual number of retries used in attempting to recover from the error condition.

Table 231. Actual Retry Count Bytes

Bit **Byte**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
15	SKSV	Reserved						
16	(MSB)	ACTUAL RETRY COUNT [1]						
17								(LSB)

[1] The ACTUAL RETRY COUNT field returns implementation specific information on the actual number of retries used in attempting to recover an error or exception condition.

Not all drives implement reporting actual retry count in bytes 15, 16, and 17. See individual drive's Product Manual, Volume 1.

Note. This field relates to the retry count fields specified within the error recovery parameters page of the MODE SELECT command (see Table 156).

If the sense key is NOT READY and the SKSV bit is one, the SENSE KEY SPECIFIC field shall be defined as shown in Table 230. These fields are only defined for the FORMAT UNIT command with the IMMED bit set to one.

Table 232. Progress Indication Bytes

Bit **Byte**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
15	SKSV	Reserved						
16	(MSB)	PROGRESS INDICATION [1]						
17								(LSB)

[1] The PROGRESS INDICATION field is a percent complete indication in which the returned value is the numerator that has 65536 (10000h) as its denominator. The progress indication shall be based upon the total operation.

Support or non-support for the Progress Indication field is given in individual drive's Product Manual, Volume 1, SCSI bus conditions and miscellaneous features supported section.

Note. It is intended that the progress indication be time related. However, since, for example, format time varies with the number of defects encountered, etc., it is reasonable for the target to assign values to various steps within the process. The granularity of these steps should be small enough to provide reasonable assurances to the initiator that progress is being made.

Table 233. Segment pointer bytes

Bit Byte	7	6	5	4	3	2	1	0
15	SKSV [1]	Reserved	SD [2]	Reserved	BPV [3]	BIT POINTER [4]		
16	(MSB)	FIELD POINTER [5]						
17								(LSB)

[1] SKSV described previously in Table 230.

[2] The Segment Descriptor (SD) bit indicates whether the field pointer is with reference to the start of the parameter list or to the start of a segment descriptor. An SD value of zero indicates that the field pointer is relative to the start of the parameter list. An SD value of one indicates that the field pointer is relative to the start of the segment descriptor indicated by the third and fourth bytes of the Command-Specific Information field of the Extended COPY command, which devices addressed by this manual do not implemented.

[3] A Bit Pointer Valid (BPV) bit of zero indicates that the value in the Bit Pointer field is not valid. A BPV bit of one indicates that the Bit Pointer field specifies which bit of the byte designated by the Field Pointer field is in error.

[4] When a multiple-bit field is in error, the BIT POINTER field shall point to the most-significant (left-most) bit of the field.

[5] The FIELD POINTER field indicates which byte of the parameter list or segment descriptor was in error.

Note. If the parameter list is in excess of 65528 bytes in length and SD is 0, the FIELD POINTER value may not fit in two bytes provided by the sense key specific format definition.

8.38.1.1 Current errors

Response code 70h (current error) indicates that the sense data returned is the result of an error or exception condition on the task that returned the CHECK CONDITION status or a protocol specific failure condition. This includes errors generated during processing of the command. It also includes errors not related to any command that are detected during processing of a command (e.g., disk servo-mechanism failure, off-track errors, or power-up test errors).

8.38.1.2 Deferred errors

Response code 71h (deferred error) indicates that the CHECK CONDITION status returned is the result of an error or exception condition that occurred during processing of a previous command for which GOOD, CONDITION MET, INTERMEDIATE, and INTERMEDIATE-CONDITION MET status has already been returned. Such commands are associated with the use of the immediate bit (start unit) and with some forms of caching, and with multiple command buffering. Targets that implement these features are required to implement deferred error reporting.

The deferred error may be indicated by returning CHECK CONDITION status to an application client accessed through the initiator port as described in this subclause. Examples of this latter type of error include disc servo-mechanism off-track errors and power-up test errors.

When the drive does not use the AEN (Asynchronous Event Notification) Feature, the deferred error may be indicated by returning CHECK CONDITION status to the appropriate initiator as described below. The subsequent execution of a REQUEST SENSE command shall return the deferred error sense information.

If CHECK CONDITION status for a deferred error is returned, the current command has not performed any storage operations or output operations to the media. After the target detects a deferred error condition on a logical unit, it shall return a deferred error according to the rules described below:

1. If a deferred error can be recovered with no external system intervention, a deferred error indication shall not be posted unless required by the error handling parameters of the MODE SELECT command. The occurrence of the error may be logged if statistical or error logging is supported.
2. If a deferred error can be associated with a causing initiator port and with a particular function or a particular subset of data, and the error is either unrecovered or required to be reported by the mode parameters, a deferred error indication shall be returned to the causing initiator through the initiator port associated with the error. If a SCSI initiator port other than the causing initiator attempts access to the particular function or subset of data associated with the deferred error, and the TST field equals 000b (see SPC-3, Section 7.4.6), the target shall respond to the command with a BUSY or ACA ACTIVE status according to the requirements in SAM-3. If an initiator accessed through an initiator port other than the initiator port associated with the error attempts access to the particular function or subset of data associated with the deferred error and the TST field equals 001b, the command attempting the access shall not be blocked by the deferred error and the cause of the deferred error may result in an error being reported for the command attempting the access.

   Note. Not all devices may be sufficiently sophisticated to identify the function or data that has failed. Those that cannot should treat the error in the following manner.
3. If a deferred error cannot be associated with a causing initiator or with a particular subset of data, a deferred error indication shall be returned on behalf of the failing logical unit to each initiator. If multiple deferred errors have accumulated for some initiators, only the last error shall be returned.
4. If the SCSI target device is unable to associate a deferred error with a particular logical unit, it shall return a deferred error indication to an initiator associated with every logical unit and shall return the deferred error through an appropriate initiator port; or
5. If a current command has not yet started executing and a deferred error occurs, the command shall be terminated with CHECK CONDITION status and deferred error information posted in the sense data. By convention, the current command is considered to have started execution if the target has changed phase from the COMMAND phase to the next normal phase of the command sequence. If a deferred error occurs while a current command is executing and the current command has been affected by the error, the command shall be terminated by CHECK CONDITION status and current error information shall be returned in the sense data. In this case, if the current error information does not adequately define the deferred error condition, a deferred error may be returned after the current error information has been recovered. If a deferred error occurs while a current command is executing and the current command completes successfully, the target may choose to return the deferred error information after the completion of the current command.

Deferred errors may indicate that an operation was unsuccessful long after the command performing the data transfer returned Good status. If data that cannot be replicated or recovered from other sources is being stored using such buffered write operations, synchronization commands should be performed before the critical data is destroyed in the host initiator. This is necessary to be sure that recovery actions can be taken if deferred errors do occur in the storing of the data. When AEN is not implemented, the synchronizing process provides the necessary commands to allow returning CHECK CONDITION status and subsequent returning of deferred error sense information after all buffered operations are guaranteed to be complete.

8.38.2 Sense Key and Sense Code descriptions

Table 234 lists the Sense Keys in the extended sense data format that are used by the drive.

Table 234. Applicable Drive Sense Keys

Sense Key	Description
0h	No Sense—Indicates there is no specific Sense Key information to be reported for the drive. This would be the case for a successful command or when the ILI bit is one.
1h	Recovered Error—Indicates the last command completed successfully with some recovery action performed by the drive. When multiple recovered errors occur, the last error that occurred is reported by the additional sense bytes. This key is also used to indicate maximum temperature threshold exceeded (see also additional sense code and sense code qualifier 0B 01). **Note:** For some Mode settings, the last command may have terminated before completing.
2h	Not Ready—Indicates the logical unit addressed cannot be accessed. Operator intervention may be required to correct this condition.
3h	Medium Error—Indicates the command terminated with a non-recovered error condition, probably caused by a flaw in the medium or an error in the recorded data.
4h	Hardware Error—Indicates the drive detected a non-recoverable hardware failure while performing the command or during a self-test. This includes SCSI interface parity error, controller failure, device failure, etc.
5h	ILLEGAL REQUEST—Indicates an illegal parameter in the command descriptor block or in the additional parameters supplied as data for some commands (FORMAT UNIT, MODE SELECT, etc.). If the drive detects an invalid parameter in the Command Descriptor Block, it shall terminate the command without altering the medium. If the drive detects an invalid parameter in the additional parameters supplied as data, the drive may have already altered the medium. This sense key may also indicate that an invalid IDENTIFY message was received. This could also indicate an attempt to write past the last logical block.
6h	UNIT ATTENTION—Indicates the drive may have been reset. See Section 7.6.6 for more detailed information about the UNIT ATTENTION condition.
7h	Data Protect—Indicates that a command that reads or writes the medium was attempted on a block that is protected from this operation. The Read or Write operation is not performed.
9h	Firmware Error—vendor-specific sense key.
Bh	Aborted Command—Indicates the drive aborted the command. The initiator may be able to recover by trying the command again.
Ch	Equal—Indicates a Search Data command has satisfied an equal comparison.
Dh	Volume Overflow—Indicates a buffered peripheral device has reached the end-of-medium partition and data remains in the buffer that has not been written to the medium.
Eh	Miscompare—Indicates that the source data did not match the data read from the medium.
Fh	Reserved

8.38.3 Additional Sense and Additional Sense Qualifier codes

Table 235 lists the Additional Sense (ASC) and Additional Sense Qualifier (ASCQ) codes. Code values are in hexadecimal. ANSI standard SPC-3 lists a more complete table in error description alphabetical order. This Table 235 list adequately covers all Seagate drives, however.

Note. Table 235 is for reference only, as not all drives covered by this manual support all of the codes listed.

Table 235. Error codes for ASC and ASCQ bytes (Table 229 bytes 12 and 13 of Sense Data)

ASC (byte 12)	ASCQ (byte 13)	Description
00	00	No Additional Sense Information
01	00	No Index/Logical Block Signal
02	00	No SEEK Complete
03	00	Peripheral Device Write Fault
03	86	Write Fault Data Corruption
04	00	Logical Unit Not Ready, Cause Not Reportable
04	01	Logical Unit Not Ready, Becoming Ready
04	02	Logical Unit Not Ready, SMART UNIT Required
04	03	Logical Unit Not Ready, Manual Intervention Required
04	04	Logical Unit Not Ready, Format in Progress
05	00	ILLEGAL REQUEST
06	00	UNIT ATTENTION
07	00	Data Protect
08	00	Logical Unit Communication Failure
08	01	Logical Unit Communication Time-Out
08	02	Logical Unit Communication Parity Error
09	00	vendor-specific Sense Key
09	01	Servo Fault
09	04	Head Select Fault
0A	00	Error Log Overflow
0B	00	Aborted Command
0B	01	Warning—Specified Temperature Exceeded
0C	00	Write Error
0C	01	Write Error Recovered With Auto-Reallocation
0C	02	Write Error—Auto Reallocation Failed
0E	00	Data Miscompare
10	00	ID CRC Or ECC Error
11	00	Unrecovered Read Error
11	01	Read Retries Exhausted
11	02	Error Too Long To Correct
11	04	Unrecovered Read Error—Auto Reallocation Failed
12	00	Address Mark Not Found For ID Field

Table 235. Error codes for ASC and ASCQ bytes (Table 229 bytes 12 and 13 of Sense Data)

ASC (byte 12)	ASCQ (byte 13)	Description
12	01	Recovered Data Without ECC Using Previous Logical Block ID
12	02	Recovered Data With ECC Using Previous Logical Block ID
14	00	Logical Block Not Found
14	01	Record Not Found
15	00	Random Positioning Error
15	01	Mechanical Positioning Error
15	02	Positioning Error Detected By Read Of Medium
16	00	Data Synchronization Mark Error
17	00	Recovered Data With No Error Correction Applied
17	01	Recovered Data Using Retries
17	02	Recovered Data Using Positive Offset
17	03	Recovered Data Using Negative Offset
17	05	Recovered Data Using Previous Logical Block ID
17	06	Recovered Data Without ECC—Data Auto Reallocated
18	00	Recovered Data With ECC
18	01	Recovered Data With ECC And Retries Applied
18	02	Recovered Data With ECC And/Or Retries, Data Auto-Reallocated
18	05	Recovered Data—Recommand Reassignment
18	06	Recovered Data Using ECC and Offsets
18	07	Recovered Data With ECC—Data Rewritten
19	00	Defect List Error
19	01	Defect List Not Available
19	02	Defect List Error In Primary List
19	03	Defect List Error in Grown List
19	0E	Fewer than 50% Defect List Copies
1A	00	Parameter List Length Error
1B	00	Synchronous Data Transfer Error
1C	00	Defect List Not Found
1C	01	Primary Defect List Not Found
1C	02	Grown Defect List Not Found
1C	83	Seagate Unique Diagnostic Code
1D	00	Miscompare During Verify Operation
1F	00	Number of Defects Overflows the Allocated Space That The Read Defect Command Can Handle
20	00	Invalid Command Operation Code
21	00	Logical Block Address Out Of Range
24	00	Invalid Field In CDB
25	00	Logical Unit Not Supported

Table 235. Error codes for ASC and ASCQ bytes (Table 229 bytes 12 and 13 of Sense Data)

ASC (byte 12)	ASCQ (byte 13)	Description
26	00	Invalid Field In Parameter List
26	01	Parameter Not Supported
26	02	Parameter Value Invalid
26	03	Invalid Field Parameter—Threshold Parameter
26	04	Invalid Release of Active Persistent Reserve
26	97	Invalid Field Parameter—TMS Firmware Tag
26	98	Invalid Field Parameter—Check Sum
26	99	Invalid Field Parameter—Firmware Tag
27	00	Write Protected
29	00	Power On, Reset, Or Bus Device Reset Occurred
29	01	Power-On Reset Occurred
29	02	SCSI Bus Reset Occurred
29	03	Bus Device Reset Function Occurred
29	04	Device Internal Reset Occurred
29	05	Transceiver Mode Changed To Single-Ended
29	06	Transceiver Mode Changed To LVD
2A	01	Mode Parameters Changed
2A	02	Log Parameters Changed
2A	03	Reservations preempted
2A	04	Reservations Released
2A	05	Registrations Preempted
2C	00	Command Sequence Error
2F	00	Tagged Commands Cleared By Another Initiator
31	00	Medium Format Corrupted
31	01	Format Failed
31	91	Format Corrupted, World Wide Name (WWN) Is Invalid
32	00	No Defect Spare Location Available
32	01	Defect List Update Error
32	02	No Spares Available—Too Many Defects On One Track
35	00	Unspecified Enclosure Services Failure
35	01	Unsupported Enclosure Function
35	02	Enclosure Services Unavailable
35	03	Enclosure Transfer Failure
35	04	Enclosure Transfer Refused
37	00	Parameter Rounded
3D	00	Invalid Bits In Identify Message
3E	00	Logical Unit Has Not Self Configured Yet
3F	00	Target Operating Conditions Have Changed

Table 235. Error codes for ASC and ASCQ bytes (Table 229 bytes 12 and 13 of Sense Data)

ASC (byte 12)	ASCQ (byte 13)	Description
3F	01	Microcode Changed
3F	02	Changed Operating Definition
3F	05	Device Identifier Changed
3F	90	Invalid APM Parameters
3F	91	World Wide Name (WWN) Mismatch
40	01	DRAM Parity Error
40	02	DRAM Parity Error
42	00	Power-On Or Self-Test Failure
43	00	Message Error
44	00	Internal Target Failure
45	00	Select/Reselection Failure
47	00	SCSI Parity Error
48	00	Initiator Detected Error Message Received
49	00	Invalid Message Error
4C	00	Logical Unit Failed Self-Configuration
4E	00	Overlapped Commands Attempted
55	01	XOR Cache is Not Available
5B1	00	Log Exception
5B*	01	Threshold Condition Met
5B*	02	Log Counter At Maximum
5B*	03	Log List Codes Exhausted
5C	00	RPL Status Change
5C	01	Spindles Synchronized
5C	02	Spindles Not Synchronized
5D	00	Failure Prediction Threshold Exceeded
5D	FF	False Failure Prediction Threshold Exceeded
65	00	Voltage Fault
80	00	General Firmware Error Qualifier
81	00	Reassign Power—Fail Recovery Failed

1. Can be supported, but is a factory installed option.

8.39 RESERVE (6) command (16h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.40 RESERVE (10) command (56h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.41 REZERO UNIT command (01h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.42 SEARCH DATA EQUAL command (31h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.43 SEARCH DATA HIGH command (30h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.44 SEARCH DATA LOW command (32h)

Obsolete. If received, a CHECK CONDITION status is sent.

8.45 SEEK (6) command (0Bh)

Obsolete. If received, a CHECK CONDITION status is sent.

8.46 SEEK EXTENDED command (2Bh)

The SEEK EXTENDED command requests that the drive seek to the specified LOGICAL BLOCK ADDRESS. This command is implemented with the drive specific parameters listed in Table 236.

Table 236. SEEK EXTENDED command (2Bh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	0	1	0	1	1
1	RESERVED [1]			Reserved				
2	(MSB)	LOGICAL BLOCK ADDRESS [2]						
:								
5								(LSB)
6	Reserved							
:								
8								
9	CONTROL [3]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] This command operates the same as the SEEK command (Section 8.45) except that a four byte LOGICAL BLOCK ADDRESS is specified.

[3] See Table 1, “CONTROL field,” in Section 7.2.1.

8.47 SEND DIAGNOSTIC command (1Dh)

The SEND DIAGNOSTIC command (see Table 237) requests the target to perform diagnostic operations on the target, on the logical unit, or on both. Targets that support this command shall implement, at a minimum, the default self-test feature (the SelfTest bit equal to one and a PARAMETER LIST LENGTH of zero). When the SelfTest bit is zero and the Self-Test Code field contains 000b, this command is usually followed by a RECEIVE DIAGNOSTIC RESULTS (see Section 8.33) command.

Table 237. SEND DIAGNOSTIC command (1Dh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	1	1	1	0	1
1	SELF-TEST CODE [1]			PF [2]	Reserved	SELF TEST [1] [3]	DEVOFFL [4]	UNIT OFFL [5]
2	Reserved							
3	(MSB)	PARAMETER LIST LENGTH [6]						
4								(LSB)
5	Control [7]							

[1] When the SELF TEST bit is one, the SELF-TEST CODE field shall contain 000b. When the SELF TEST bit is zero, the contents of the SELF-TEST CODE field are as specified in Table 238.

Table 238. Self-Test Code field values

Value	Function name	Description
000b	N/A	This value shall be used when the Self Test bit is set to one or if the SEND DIAGNOSTIC command is not invoking one of the other self-test functions such as enclosure services (see ANSI SES, T10/1212D) or the Translate Address page (see Section 8.47.2).
001b	Background short self-test	The target shall start its short self-test (see Section 9.2.2) in the background mode (see Section 9.2.3.2). The PARAMETER LIST LENGTH field shall contain zero.
010b	Background extended self-test	The target shall start its extended self-test (see Section 9.2.2) in the background mode (see Section 9.2.3.2). The PARAMETER LIST LENGTH field shall contain zero.
011b	Reserved.	
100b	Abort background self-test	The target shall abort the current self-test running in background mode. The PARAMETER LIST LENGTH field shall contain zero. This value is only valid if a previous SEND DIAGNOSTIC command specified a background self-test function and that self-test has not completed. If either of these conditions is not met, then the target shall return a CHECK CONDITION status with a sense key of ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB.
101b	Foreground short self-test	The target shall start its short self-test (see Section 9.2.2) in the foreground mode (see Section 9.2.3.1). The PARAMETER LIST LENGTH field shall contain zero.
110b	Foreground extended self-test	The target shall start its extended self-test (see Section 9.2.2) in the foreground mode (see Section 9.2.3.1). The PARAMETER LIST LENGTH field shall contain zero.
111b	Reserved.	

[2] A Page Format (PF) bit of one specifies that the SEND DIAGNOSTIC parameters conform to the page structure as specified in this manual. The implementation of the PF bit is optional. See Section 8.1 in SPC-3, T10/1416D, for the definition of diagnostic pages. A PF bit of zero indicates that all SEND DIAGNOSTIC parameters are vendor-specific.

[3] A SELF TEST bit of one directs the target to complete the target's default self-test. If the self-test successfully passes, the command shall be terminated with Good status; otherwise, the command shall be terminated with CHECK CONDITION status and the sense key shall be set to Hardware Error.

A SELF TEST bit of zero requests that the target perform the diagnostic operation specified by the Self-Test Code field or in the parameter list. The diagnostic operation might or might not require the target to return parameter data that contains diagnostic results. If the return of parameter data is not required, the return of GOOD status indicates successful completion of the diagnostic operation. If the return of parameter data is required, the target shall either:

[a] perform the requested diagnostic operation, prepare the parameter data to be returned and indicate completion by returning Good status. The initiator issues a RECEIVE DIAGNOSTIC RESULTS command to recover the parameter data; or

[b] accept the parameter list, and if no errors are detected in the parameter list, return GOOD status. The requested diagnostic operation and the preparation of the parameter data to be returned are performed upon receipt of a RECEIVE DIAGNOSTIC RESULTS command.

[4] A DEVOFFL bit of one grants permission to the target to perform diagnostic operations that may affect all the logical units on a SCSI target port, e.g., alteration of reservations, log parameters, or sense data. The implementation of the DEVOFFL bit is optional. A DEVOFFL bit of zero prohibits diagnostic operations

that may be detected by subsequent tasks. When the SELF TEST bit is zero, the DEVOFFL bit shall be ignored.

[5] A UNITOFFL bit of one grants permission to the target to perform diagnostic operations that may affect the user accessible medium on the logical unit, e.g., write operations to the user accessible medium, or repositioning of the medium on sequential access devices. The implementation of the UNITOFFL bit is optional. A UNITOFFL bit of zero prohibits any diagnostic operations that may be detected by subsequent tasks. When the SELF TEST bit is zero, the UNITOFFL bit shall be ignored.

[6] The PARAMETER LIST LENGTH field specifies the length in bytes of the parameter list that shall be transferred from the initiator to the target. A PARAMETER LIST LENGTH of zero indicates that no data shall be transferred. This condition shall not be considered an error. If the specified PARAMETER LIST LENGTH results in the truncation of one or more pages (PF bit set to one) the target shall return CHECK CONDITION status with a sense key of ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB.

Note. To ensure that the diagnostic command information is not destroyed by a command sent from another initiator, either the SEND DIAGNOSTIC command should be linked to the RECEIVE DIAGNOSTIC RESULTS command or the logical unit should be reserved.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

[8] The FUNCTION CODE field values are given in Table 238.

8.47.1 Supported Diagnostic page—SEND DIAGNOSTIC

The Supported Diagnostic page (Table 239) instructs the drive to make available the list of all supported diagnostic pages to be returned by a subsequent RECEIVE DIAGNOSTIC RESULTS command. The definition of this page for the SEND DIAGNOSTIC command includes only the first four bytes (Receive Diagnostic version given in Table 219). If the PAGE LENGTH field is not zero, the drive shall terminate the SEND DIAGNOSTIC command with a CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST with an additional sense code of Invalid Field In Parameter List.

Table 239. Supported Diagnostic page

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (00h)							
1	Reserved							
2	(MSB)			PAGE LENGTH (must be zero)				
3								(LSB)

8.47.2 Translate Address page—SEND DIAGNOSTIC

The Translate Address page (Table 240) allows the initiator to translate a LOGICAL BLOCK ADDRESS into a physical sector address or a physical sector into a LOGICAL BLOCK ADDRESS. The address to be translated is passed to the drive with a SEND DIAGNOSTIC command and the results are returned to the initiator during the DATA IN phase following the RECEIVE DIAGNOSTIC RESULTS command. The translated address is returned in the Translate Address page returned after the RECEIVE DIAGNOSTIC RESULTS command (see Table 217).

Table 240. Translate Address page

Bit Byte	7	6	5	4	3	2	1	0
0	PAGE CODE (40h)							
1	Reserved							
2	(MSB)			PAGE LENGTH (000Ah)				
3								(LSB)
4	Reserved					SUPPLIED FORMAT [1]		
5	Reserved					TRANSLATE FORMAT [2]		
6 : 13	ADDRESS TO TRANSLATE [3]							

[1] The SUPPLIED FORMAT field specifies the format of the ADDRESS TO TRANSLATE field. The valid values for this field are 000 for LOGICAL BLOCK ADDRESS format or 101 for physical sector address format. If the drive does not support the requested format, it shall terminate the SEND DIAGNOSTIC command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and an additional sense code shall be set to INVALID FIELD IN PARAMETER LIST.

[2] The TRANSLATE FORMAT field specifies the format to which the initiator would like the address to be translated. The valid values for this field are 000 for LOGICAL BLOCK ADDRESS format or 101 for physical sector address format. The TRANSLATE FORMAT field must be different than the Supplied Format field. If the drive does not support the requested format it shall terminate the command with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and an additional sense code shall be set to INVALID FIELD IN PARAMETER LIST.

[3] The ADDRESS TO TRANSLATE field contains a single address the initiator is requesting the drive to translate. The format of this field is defined by the SUPPLIED FORMAT field. The supported formats are shown in Table 221 and Table 222.

For systems which support disconnection, the drive will disconnect while executing this command.

8.48 SET DEVICE IDENTIFIER command (A4h)

The SET DEVICE IDENTIFIER command (see Table 241) requests that the device identifier information in the logical unit be set to the value received in the SET DEVICE IDENTIFIER parameter list. This command is optional for all device types. As defined in the SCSI Controller Command Set-2 (SCC-2), T10/1225D, the SET DEVICE IDENTIFIER command is the Set Peripheral Device/Component Device Identifier service action of the Maintenance Out command. Additional Maintenance Out service actions are defined in SCC-2. Only the SET DEVICE IDENTIFIER operation code and service action concerns all SCSI devices.

On successful completion of a SET DEVICE IDENTIFIER command a UNIT ATTENTION shall be generated for all initiators except the one that issued the service action. When reporting the UNIT ATTENTION condition the additional sense code shall be set to DEVICE IDENTIFIER CHANGED.

Table 241. SET DEVICE IDENTIFIER command (A4h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	0	0	1	0	0
1	Reserved			SERVICE ACTION (06h) [1]				
2	Reserved							
3								
4	See SCC-2 [2]							
5								
6	(MSB)	PARAMETER LIST LENGTH [3]						
:								
9								(LSB)
10	Reserved						See SCC-2 [2]	Reserved
11	CONTROL [4]							

[1] SERVICE ACTION must be set to 06h.

[2] SCSI Controller Command Set-2 (SCC-2), T10/1225D, defines specific usages for bytes 4 and 5, and bit 1 in byte 10. However these fields are reserved for the SET DEVICE IDENTIFIER command defined by this manual.

[3] The PARAMETER LIST LENGTH field specifies the length in bytes of the Identifier that shall be transferred from the initiator to the target. The maximum value for this field shall be 512 bytes. A PARAMETER LIST LENGTH of zero indicates that no data shall be transferred, and that subsequent REPORT DEVICE IDENTIFIER commands shall return an Identifier length of zero. Logical units that implement this command shall be capable of accepting a PARAMETER LIST LENGTH of 64 bytes or less. If the PARAMETER LIST LENGTH exceeds 64 bytes and the logical unit is not capable of storing the requested number of bytes, then the target shall return CHECK CONDITION status with the sense key set to ILLEGAL REQUEST and an additional sense code of Invalid Field in CDB.

[4] See Table 1, “CONTROL field,” in Section 7.2.1.

The SET DEVICE IDENTIFIER parameter list (see Table 242) contains the identifier to be set by the addressed logical unit.

Table 242. SET DEVICE IDENTIFIER parameter list

Bit Byte	7	6	5	4	3	2	1	0
0	IDENTIFIER [1]							
n								

[1] The IDENTIFIER field shall be a vendor-specific value, to be returned in subsequent REPORT DEVICE IDENTIFIER commands.

8.49 SET LIMITS command (33h)

Not implemented. If received, a CHECK CONDITION status is sent.

8.50 START/STOP UNIT command (1Bh)

The START/STOP UNIT command requests that the target enable the logical unit for further operations (start) or stop spindle rotation (stop).

The drive is able to execute the following commands when the drive spindle is not rotating, or in a simulated stopped condition:

TEST UNIT READY
REQUEST SENSE
INQUIRY
RESERVE
RELEASE
MOTOR START
RECEIVE DIAGNOSTIC RESULTS
WRITE DATA BUFFER
READ DATA BUFFER

The remaining commands (see individual drive's Product Manual, Volume 1, SCSI interface commands supported section, for a list of all commands supported) cannot be executed until after the drive has spindled up. If the drive receives one of these commands before it can be executed, a CHECK CONDITION status is returned (with sense key of NOT READY).

For systems that support disconnection, the drive disconnects when a START UNIT procedure is commanded, and reconnects when the unit is up to speed and ready or when operation is initiated if the IMMED bit is one.

Table 243. START/STOP UNIT command (1Bh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	1	1	0	1	1
1	Reserved							IMMED [1]
2	Reserved							
3								
4	Reserved							START [2]
5	CONTROL [3]							

[1] An Immediate (IMMED) bit of zero indicates that status shall be returned after the operation is completed. If the IMMED bit is one, status is returned as soon as the operation is initiated.

[2] When the START bit is one, this requests that the logical unit be made ready for use. If the START bit is zero (requesting that the unit be stopped), the request causes the drive to either actually stop the spindle or to simulate the drive spindle stopped condition. In this latter situation, the drive spindle actually continues to spin and the drive reports NOT READY in response to media access commands. Power consumption is as when in idle mode. The type of stop implemented is given in individual drive's Product Manual, Volume 1.

[3] See Table 1, "CONTROL field," in Section 7.2.1.

8.51 SYNCHRONIZE CACHE (10) command (35h)

The SYNCHRONIZE CACHE (10) command (Table 244) ensures that logical blocks in the cache memory, within the specified range, have their most recent data value recorded on the physical medium. If a more recent data value for a logical block within the specified range exists in the cache memory than on the physical medium, then the logical block from the cache memory shall be written to the physical medium. Logical blocks are not necessarily removed from the cache memory as a result of the Synchronize Cache operation.

Table 244. SYNCHRONIZE CACHE (10) command (35h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	1	0	1	0	1
1	0	0	0	Reserved			IMMED [2]	RELADR [3]
	RESERVED [1]							
2 : 5	(MSB)			LOGICAL BLOCK ADDRESS [4]				(LSB)
6	Reserved							
7 8	(MSB)			NUMBER OF BLOCKS [5]				(LSB)
9	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] An Immediate (IMMED) bit of one indicates that the drive shall return status as soon as the command descriptor block has been validated. An IMMED bit of zero indicates that the status shall not be returned until the operation has been completed. If the IMMED bit is one and the target does not support it, the command shall terminate with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN CDB.

[3] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first RELADR logical block of the range of logical blocks to be operated on by this command.

[4] The LOGICAL BLOCK ADDRESS field specifies the logical block at which the Synchronize Cache operation begins.

[5] The NUMBER OF BLOCKS field specifies the total number of contiguous logical blocks within the range. A number of blocks of zero indicates that all remaining logical blocks on the logical unit shall be within the range.

A logical block within the specified range that is not in cache memory is not considered an error. Multiple locks may be in effect from more than one initiator. Locks from different initiators may overlap. An unlock of an overlapped area does not release the lock of another initiator.

[6] See Table 1, "CONTROL field," in Section 7.2.1.

8.52 SYNCHRONIZE CACHE (16) command (91h)

The SYNCHRONIZE CACHE (16) command (Table 244) ensures that logical blocks in the cache memory, within the specified range, have their most recent data value recorded on the physical medium. If a more recent data value for a logical block within the specified range exists in the cache memory than on the physical medium, then the logical block from the cache memory shall be written to the physical medium. Logical blocks are not necessarily removed from the cache memory as a result of the Synchronize Cache operation.

Table 245. SYNCHRONIZE CACHE (16) command (91h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	0	1	0	0	0	1
1	0	0	0	Reserved			IMMED [2]	RELADR [3]
	RESERVED [1]							
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [4]							
9								(LSB)
10	(MSB)							
:	NUMBER OF BLOCKS [5]							
13								(LSB)
14	Reserved							
15	CONTROL [6]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] An Immediate (IMMED) bit of one indicates that the drive shall return status as soon as the command descriptor block has been validated. An IMMED bit of zero indicates that the status shall not be returned until the operation has been completed. If the IMMED bit is one and the target does not support it, the command shall terminate with CHECK CONDITION status. The sense key shall be set to ILLEGAL REQUEST and the additional sense code shall be set to INVALID FIELD IN CDB.

[3] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first RELADR logical block of the range of logical blocks to be operated on by this command.

[4] The LOGICAL BLOCK ADDRESS field specifies the logical block at which the Synchronize Cache operation begins.

[5] The NUMBER OF BLOCKS field specifies the total number of contiguous logical blocks within the range. A number of blocks of zero indicates that all remaining logical blocks on the logical unit shall be within the range.

A logical block within the specified range that is not in cache memory is not considered an error. Multiple locks may be in effect from more than one initiator. Locks from different initiators may overlap. An unlock of an overlapped area does not release the lock of another initiator.

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

8.53 TEST UNIT READY command (00h)

The TEST UNIT READY command provides a means to verify the logical unit is ready. This is not a request for a self-test. If the logical unit (drive) can accept an appropriate medium access command without returning CHECK CONDITION status, the drive returns a GOOD status. See Table 246 for proper format.

Table 246. TEST UNIT READY command (00h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0
1	RESERVED [1]			Reserved				
2	Reserved							
:								
4								
5	CONTROL [2]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] See Table 1, “CONTROL field,” in Section 7.2.1.

If the drive cannot become operational or is in a state such that a SCSI initiator port action (e.g., Start command) is required to make the unit ready, the drive returns CHECK CONDITION status with a Sense Key of NOT READY. One of several possible additional sense codes indicates the reason for the NOT READY condition.

8.54 VERIFY (10) command (2Fh)

The VERIFY (10) command requests that the target verify the data written on the medium. This command is implemented with the drive specific parameters listed in Table 247. The drive disconnects while this command is being executed if the initiator supports disconnect/reconnect.

Table 247. VERIFY (10) command (2Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	0	1	1	1	1
1	RESERVED [1]			DPO [2]	Reserved	BLKVFY [3]	BYTCHK [4]	RELADR [5]
2	(MSB)	LOGICAL BLOCK ADDRESS [6]						
:								
5								(LSB)
6	Reserved							
7	(MSB)	VERIFICATION LENGTH [7]						
8								(LSB)
9	CONTROL [8]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the logical blocks accessed by this command are likely to be accessed again in the near future.

[3] For direct access block devices, the Blank Verify (BLKVFY) bit shall be considered reserved.

[4] A Byte Check (BYTCHK) bit of zero causes the verification to be simply a medium verification (CRC, ECC, etc.). A BYTCHK bit of one causes a byte by byte compare of data on the medium and the data transferred from the initiator. If the compare is unsuccessful, the command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set to MISCOMPARE.

[5] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.
A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[6] The LOGICAL BLOCK ADDRESS specifies the logical block at which the VERIFY operation shall begin, if RELADR is zero (see note [5]).

[7] The VERIFICATION LENGTH specifies the number of contiguous logical blocks of data that shall be verified. A VERIFICATION LENGTH of zero indicates that no logical blocks shall be verified (an Implied SEEK is still performed). This condition shall not be considered an error. Any other value indicates the number of

logical blocks that shall be verified. The VERIFICATION LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[8] See Table 1, “CONTROL field,” in Section 7.2.1.

8.55 VERIFY (12) command (AFh)

The VERIFY (12) command requests that the target verify the data written on the medium. This command is implemented with the drive specific parameters listed in Table 248. The drive disconnects while this command is being executed if the initiator supports disconnect/reconnect.

Table 248. VERIFY (12) command (AFh)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	0	1	1	1	1
1	RESERVED [1]			DPO [2]	Reserved	BLKVFY [3]	BYTCHK [4]	RELADR [5]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [6]							
5								(LSB)
6	(MSB)							
:	VERIFICATION LENGTH [7]							
9								(LSB)
10	Reserved							
11	CONTROL [8]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the logical blocks accessed by this command are likely to be accessed again in the near future.

[3] For direct access block devices, the Blank Verify (BLKVFY) bit shall be considered reserved.

[4] A Byte Check (BYTCHK) bit of zero causes the verification to be simply a medium verification (CRC, ECC, etc.). A BYTCHK bit of one causes a byte by byte compare of data on the medium and the data transferred from the initiator. If the compare is unsuccessful, the command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set to MISCOMPARE.

[5] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[6] The LOGICAL BLOCK ADDRESS specifies the logical block at which the VERIFY operation shall begin, if RELADR is zero (see note [5]).

[7] The VERIFICATION LENGTH specifies the number of contiguous logical blocks of data that shall be verified. A VERIFICATION LENGTH of zero indicates that no logical blocks shall be verified (an Implied SEEK is still performed). This condition shall not be considered an error. Any other value indicates the number of logical blocks that shall be verified.

[8] See Table 1, “CONTROL field,” in Section 7.2.1.

8.56 VERIFY (16) command (8Fh)

The VERIFY (16) command requests that the target verify the data written on the medium. This command is implemented with the drive specific parameters listed in Table 249. The drive disconnects while this command is being executed if the initiator supports disconnect/reconnect.

Table 249. VERIFY (16) command (8Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	0	0	1	1	1	1
1	RESERVED [1]			DPO [2]	Reserved	BLKVFY [3]	BYTCHK [4]	RELADR [5]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [6]							
9								(LSB)
10	(MSB)							
:	VERIFICATION LENGTH [7]							
13								(LSB)
14	Reserved							
15	CONTROL [8]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the logical blocks accessed by this command are likely to be accessed again in the near future.

[3] For direct access block devices, the Blank Verify (BLKVFY) bit shall be considered reserved.

[4] A Byte Check (BYTCHK) bit of zero causes the verification to be simply a medium verification (CRC, ECC, etc.). A BYTCHK bit of one causes a byte by byte compare of data on the medium and the data transferred from the initiator. If the compare is unsuccessful, the command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set to MISCOMPARE.

[5] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[6] The LOGICAL BLOCK ADDRESS specifies the logical block at which the VERIFY operation shall begin, if RELADR is zero (see note [5]).

[7] The VERIFICATION LENGTH specifies the number of contiguous logical blocks of data that shall be verified. A VERIFICATION LENGTH of zero indicates that no logical blocks shall be verified (an Implied SEEK is still performed). This condition shall not be considered an error. Any other value indicates the number of logical blocks that shall be verified.

[8] See Table 1, “CONTROL field,” in Section 7.2.1.

8.57 WRITE (6) command (0Ah)

The WRITE (6) command (Table 250) requests that the drive write, to the medium, the data transferred by the initiator.

Table 250. WRITE (6) command (0Ah)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	1	0
1	RESERVED [1]			(MSB)	LOGICAL BLOCK ADDRESS [2]			
2	LOGICAL BLOCK ADDRESS							
3								(LSB)
4	TRANSFER LENGTH [3]							
5	CONTROL [4]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin.

[3] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data to be transferred. A TRANSFER LENGTH of zero indicates that 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks that shall be transferred. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[4] See Table 1, “CONTROL field,” in Section 7.2.1.

For systems that support disconnection, the drive disconnects when any internal error recovery procedure is required, or the data transfer with the initiator is interrupted for more than 1 millisecond, or if the drive’s internal data buffer is full. After a disconnect, the drive reconnects depending on the value of the Buffer Empty Ratio in Page 2 of MODE SELECT Data (see Section 8.11).

The initiator must send requested write data to the drive until the drive sends Completion status during a STATUS phase or until the initiator Resets/Aborts the command or clears the queue.

Note. The drive may disconnect and reconnect while executing this command.

Sense Data is valid after this command is executed and Completion status is sent (refer to the READ command description in Section 8.21).

If the RCD bit is set to zero on the Caching Mode page 08h (cache is enabled), the data that is written by this command remains in the cache buffer, if no write errors are encountered. This allows a READ command to access the same data from the cache buffer instead of accessing the media, if the same LBA is requested by the READ command.

This command is terminated with a RESERVATION CONFLICT status and no data is written if any reservation access conflict (see Section 8.39) exists.

If any of the following conditions occur, this command is terminated with a CHECK CONDITION status, and if extended sense is implemented, the Sense Key is set as indicated in the following table. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	**Sense Key**
INVALID LOGICAL BLOCK ADDRESS	VOLUME OVERFLOW (see note below)
TARGET RESET SINCE LAST COMMAND FROM THIS INITIATOR	UNIT ATTENTION

Note. The extended sense Information Bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.58 WRITE (10) command (2Ah)

The WRITE (10) command requests that the drive write to the medium the data transferred by the initiator. This command is implemented with the drive specific parameters listed in Table 251. Refer also to individual drive's Product Manual, Volume 1, Write Caching section, for information on Write Cache Control.

Table 251. WRITE (10) command (2Ah)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	0	1	0	1	0
1	RESERVED [1]			DPO [2]	FUA [3]	EBP	Reserved	RELADR [4]
2	(MSB)							
3	LOGICAL BLOCK ADDRESS [5]							
4								
5								(LSB)
6	Reserved							
7	(MSB)	TRANSFER LENGTH [6]						
8								(LSB)
9	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (Caching enabled).

[3] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin, if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A TRANSFER LENGTH of zero indicates no logical blocks shall be transferred. This condition shall not be considered an error and no data shall be written. Any other value indicates the number of logical blocks that shall be transferred. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[7] See Table 1, "CONTROL field," in Section 7.2.1.

This command operates the same as the WRITE command (Section 8.57) except that in the CDB for this command a four byte LOGICAL BLOCK ADDRESS and a two byte TRANSFER LENGTH may be specified.

This command shall be terminated with a RESERVATION CONFLICT status if any reservation access conflict (see Section 8.39) exists, and no data shall be written.

If any of the following conditions occur, this command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set as indicated in the following table. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	**Sense Key**
INVALID LOGICAL BLOCK ADDRESS	VOLUME OVERFLOW (see note)
TARGET RESET SINCE THE LAST COMMAND FROM THIS INITIATOR	UNIT ATTENTION

Note. The extended sense information bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.59 WRITE (12) command (AAh)

The WRITE (12) command requests that the drive write to the medium the data transferred by the initiator. This command is implemented with the drive specific parameters listed in Table 252. Refer also to individual drive's Product Manual, Volume 1, Write Caching section, for information on Write Cache Control.

Table 252. WRITE (12) command (AAh)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	0	1	0	1	0
1	RESERVED [1]			DPO [2]	FUA [3]	RESERVED	RESERVED	RELADR [4]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [5]							
5								(LSB)
6	(MSB)							
:	TRANSFER LENGTH (MSB) [6]							
9								(LSB)
10	RESERVED							
11	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (Caching enabled).

[3] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin, if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A TRANSFER LENGTH of zero indicates no logical blocks shall be transferred. This condition shall not be considered an error and no data shall be written. Any other value indicates the number of logical blocks that shall be transferred. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[7] See Table 1, "CONTROL field," in Section 7.2.1.

This command operates the same as the WRITE command (Section 8.57) except that in the CDB for this command a four byte LOGICAL BLOCK ADDRESS and a four byte TRANSFER LENGTH may be specified.

This command shall be terminated with a RESERVATION CONFLICT status if any reservation access conflict (see Section 8.39) exists, and no data shall be written.

If any of the following conditions occur, this command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set as indicated in the following table. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	**Sense Key**
INVALID LOGICAL BLOCK ADDRESS	VOLUME OVERFLOW (see note)
TARGET RESET SINCE THE LAST COMMAND FROM THIS INITIATOR	UNIT ATTENTION

Note. The extended sense information bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.60 WRITE (16) command (8Ah)

The WRITE (16) command requests that the drive write to the medium the data transferred by the initiator. This command is implemented with the drive specific parameters listed in Table 253. Refer also to individual drive’s Product Manual, Volume 1, Write Caching section, for information on Write Cache Control.

Table 253. WRITE (16) command (8Ah)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	0	0	1	0	1	0
1	RESERVED [1]			DPO [2]	FUA [3]	RESERVED	RESERVED	RELADR [4]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [5]							
9								(LSB)
10	(MSB)							
:	TRANSFER LENGTH [6]							
13								(LSB)
14	RESERVED							
15	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (Caching enabled).

[3] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin, if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A TRANSFER LENGTH of zero indicates no logical blocks shall be transferred. This condition shall not be considered an error and no data shall be written. Any other value indicates the number of logical blocks that shall be transferred. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

This command operates the same as the WRITE command (Section 8.57) except that in the CDB for this command a eight byte LOGICAL BLOCK ADDRESS and a four byte TRANSFER LENGTH may be specified.

This command shall be terminated with a RESERVATION CONFLICT status if any reservation access conflict (see Section 8.39) exists, and no data shall be written.

If any of the following conditions occur, this command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set as indicated in the following table. This table does not provide an exhaustive enumeration of all conditions that may cause the CHECK CONDITION status.

Condition	**Sense Key**
INVALID LOGICAL BLOCK ADDRESS	VOLUME OVERFLOW (see note)
TARGET RESET SINCE THE LAST COMMAND FROM THIS INITIATOR	UNIT ATTENTION

Note. The extended sense information bytes shall be set to the LOGICAL BLOCK ADDRESS of the first invalid address.

8.61 WRITE AND VERIFY (10) command (2Eh)

The WRITE AND VERIFY (10) command (Table 254) requests that the target write the data transferred from the initiator to the medium and then verify that the data is correctly written. The data is only transferred once from the initiator to the drive.

Table 254. WRITE AND VERIFY (10) command (2Eh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	0	1	1	1	0
1	RESERVED [1]			DPO [2]	Reserved		BYT CHK [3]	RELADR [4]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [5]							
5								(LSB)
6	Reserved							
7	(MSB)							
8	TRANSFER LENGTH [6]							(LSB)
9	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the logical blocks accessed by this command are likely to be accessed again in the near future.

[3] A Byte Check (BYTCHK) bit of zero causes the verification to be simply a medium verification (ECC) with no data comparison. A BYTCHK bit of one causes a byte-by-byte compare of data written on the peripheral device and the data transferred from the initiator. If the compare is unsuccessful, the command shall be terminated with a CHECK CONDITION status and the sense key shall be set to Miscompare.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A transfer length of zero indicates that no logical blocks shall be transferred. This condition shall not be considered as an error and no data shall be written. Any other value indicates the number of logical blocks that shall be transferred.

For systems that support disconnection, the drive disconnects during the execution of this command.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

8.62 WRITE AND VERIFY (12) command (AEh)

The WRITE AND VERIFY (12) command (Table 255) requests that the target write the data transferred from the initiator to the medium and then verify that the data is correctly written. The data is only transferred once from the initiator to the drive.

Table 255. WRITE AND VERIFY (12) command (AEh)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	1	0	1	1	1	0
1	RESERVED [1]			DPO [2]	Reserved		BYTCHK [3]	RELADR [4]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [5]							
5								(LSB)
6	(MSB)							
:	TRANSFER LENGTH [6]							
9								(LSB)
10	Reserved							
11	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the logical blocks accessed by this command are likely to be accessed again in the near future.

[3] A Byte Check (BYTCHK) bit of zero causes the verification to be simply a medium verification (ECC) with no data comparison. A BYTCHK bit of one causes a byte-by-byte compare of data written on the peripheral device and the data transferred from the initiator. If the compare is unsuccessful, the command shall be terminated with a CHECK CONDITION status and the sense key shall be set to Miscompare.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two’s complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A transfer length of zero indicates that no logical blocks shall be transferred. This condition shall

not be considered as an error and no data shall be written. Any other value indicates the number of logical blocks that shall be transferred.

For systems that support disconnection, the drive disconnects during the execution of this command.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

8.63 WRITE AND VERIFY (16) command (8Eh)

The WRITE AND VERIFY (16) command (Table 256) requests that the target write the data transferred from the initiator to the medium and then verify that the data is correctly written. The data is only transferred once from the initiator to the drive.

Table 256. WRITE AND VERIFY (16) command (8Eh)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	0	0	1	1	1	0
1	RESERVED [1]			DPO [2]	Reserved		BYT CHK [3]	RELADR [4]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [5]							
9								(LSB)
10	(MSB)							
:	TRANSFER LENGTH [6]							
13								(LSB)
14	RESERVED							
15	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Disable Page Out (DPO) bit of one indicates that the target shall assign the logical blocks accessed by this command the lowest priority for being fetched into or retained by the cache.

The DPO bit is used to control replacement of logical blocks in the cache memory when the host has information on the future usage of the logical blocks. If the DPO bit is set to one, the logical blocks accessed by the command are not likely to be accessed again in the near future and should not be put in the cache memory nor retained by the cache memory. If the DPO bit is zero, the logical blocks accessed by this command are likely to be accessed again in the near future.

[3] A Byte Check (BYTCHK) bit of zero causes the verification to be simply a medium verification (ECC) with no data comparison. A BYTCHK bit of one causes a byte-by-byte compare of data written on the peripheral device and the data transferred from the initiator. If the compare is unsuccessful, the command shall be terminated with a CHECK CONDITION status and the sense key shall be set to Miscompare.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin if RELADR bit is zero (see note [4]).

[6] The TRANSFER LENGTH specifies the number of contiguous logical blocks of data that shall be transferred. A transfer length of zero indicates that no logical blocks shall be transferred. This condition shall

not be considered as an error and no data shall be written. Any other value indicates the number of logical blocks that shall be transferred.

For systems that support disconnection, the drive disconnects during the execution of this command.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

8.64 WRITE BUFFER command (3Bh)

The WRITE BUFFER command may be used in conjunction with the READ BUFFER command as a diagnostic function for testing the drive's data buffer memory and the SCSI bus integrity. When used in a diagnostic mode, the medium shall not be accessed during the execution of this command. Additional modes are provided for downloading and saving executable microcode.

The function of this command and the meaning of the fields within the Command Descriptor Block depend on the mode field (Byte 1, data bits 0, 1, 2). See note [2] of Table 257.

Table 257. WRITE BUFFER command (3Bh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	1	1	0	1	1
1	Reserved [1]			MODE [2]				
2	BUFFER ID [5]							
3	(MSB)							
4				BUFFER OFFSET [5]				
5								(LSB)
6	(MSB)							
7				BYTE TRANSFER LENGTH [3]				
8								(LSB)
9	CONTROL [4]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] The MODE field is defined in the following table and in the referenced sections.

Check with individual drive’s Product Manual, Volume 1, to see which modes are supported by the drive in question.

DB3	DB2	DB1	DB0	Mode definition
0	0	0	0	Write combined header and data (Section 8.64.1)
0	0	0	1	vendor-specific (Section 8.64.2)
0	0	1	0	Write data (Section 8.64.3)
0	1	0	0	Download microcode (Section 8.64.4)
0	1	0	1	Download microcode and save (Section 8.64.5)
0	1	1	0	Download microcode with offsets (Section 8.64.6)
0	1	1	1	Download microcode with offsets and save (Section 8.64.7)
1	0	1	0	Write data to echo buffer (Section 8.64.8)
1	0	0	0	Reserved
1	0	0	1	Reserved
1	0	1	1	Reserved
:	:	:	:	Reserved
1	1	1	1	Reserved

[3] Use a READ DATA BUFFER command with mode bits set to 0011b to get the drive buffer capacity to use with the WRITE BUFFER command.

[4] See Table 1, “CONTROL field,” in Section 7.2.1. All bits are zero for all modes.

[5] These bytes used only for modes 0110b and 0111b. See sections 8.64.6 and 8.64.7 for descriptions of their usage in these modes.

8.64.1 Combined header and data mode (0000b)

In this mode, data to be transferred is preceded by a four-byte header. The four-byte header consists of all reserved bytes. The Buffer ID and the Buffer Offset fields shall be zero. The PARAMETER LIST LENGTH field specifies the maximum number of bytes that shall be transferred from the Data Out Buffer. This number includes four bytes of header, so the data length to be stored in the target’s buffer is PARAMETER LIST LENGTH minus four. The initiator should attempt to ensure that the PARAMETER LIST LENGTH is not greater than four plus the Buffer Capacity field value (see Section 8.25.1) that is returned in the header of the READ BUFFER command (mode 00b). If the PARAMETER LIST LENGTH exceeds the buffer capacity the target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST.

8.64.2 vendor-specific mode (0001b)

In this mode, the meaning of the Buffer ID, Buffer Offset, and PARAMETER LIST LENGTH fields are not specified by this manual.

8.64.3 Data mode (0010b)

In this mode, the Data Out Buffer contains buffer data destined for the logical unit. The Buffer ID field identifies a specific buffer within the logical unit. The vendor assigns buffer ID codes to buffers within the logical unit. Buffer ID zero shall be supported. If more than one buffer is supported, additional buffer ID codes shall be assigned contiguously, beginning with one. If an unsupported buffer ID code is selected, the target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

Data are written to the logical unit buffer starting at the location specified by the Buffer Offset field. The application client should conform to the offset boundary requirements returned in the READ BUFFER descriptor. If the target is unable to accept the specified buffer offset, it shall return CHECK CONDITION status and it shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

The PARAMETER LIST LENGTH field specifies the maximum number of bytes that shall be transferred from the Data Out Buffer to be stored in the specified buffer beginning at the buffer offset. The initiator should attempt to ensure that the PARAMETER LIST LENGTH plus the buffer offset does not exceed the capacity of the specified buffer. (The capacity of the buffer may be determined by the Buffer Capacity field in the READ BUFFER descriptor.) If the Buffer Offset and PARAMETER LIST LENGTH fields specify a transfer in excess of the buffer capacity, the target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

8.64.4 Download microcode mode (0100b)

If the logical unit is unable to accept this command because of some device condition, the target shall terminate each WRITE BUFFER command with this mode (100b) with a CHECK CONDITION status, a sense key of ILLEGAL REQUEST, and shall set the additional sense code to Command Sequence Error.

In this mode, vendor-specific microcode or control information shall be transferred to the control memory space of the logical unit. After a power-cycle or reset, the device operation shall revert to a vendor-specific condition. The meanings of the Buffer ID, Buffer Offset, and PARAMETER LIST LENGTH fields are not specified by this manual and are not required to be zero-filled. When the microcode download has completed successfully the target shall generate a unit attention condition for all initiators except the one that issued the WRITE BUFFER command (see SAM-2). The additional sense code shall be set to Microcode Has Been Changed.

8.64.5 Download microcode and save mode (0101b)

If the logical unit is unable to accept this command because of some device condition, the target shall terminate each WRITE BUFFER command with this mode (101b) with a CHECK CONDITION status, a sense key of ILLEGAL REQUEST, and shall set the additional sense code to Command Sequence Error.

In this mode, vendor-specific microcode or control information shall be transferred to the logical unit and, if the WRITE BUFFER command is completed successfully, also shall be saved in a non-volatile memory space (semiconductor, disk, or other). The downloaded code shall then be effective after each power-cycle and reset until it is supplanted in another download microcode and save operation. The meanings of the Buffer ID, Buffer Offset, and PARAMETER LIST LENGTH fields are not specified by this manual and are not required to be zero-filled. When the download microcode and save command has completed successfully the target shall generate a unit attention condition (see SAM-2) for all initiators except the one that issued the WRITE BUFFER command. When reporting the unit attention condition, the target shall set the additional sense code to Microcode Has Been Changed.

8.64.6 Download microcode with offsets (0110b)

In this mode, the initiator may split the transfer of the vendor-specific microcode or control information over two or more WRITE BUFFER commands. If the logical unit is unable to accept this command because of some device condition, the target shall terminate each WRITE BUFFER command with this mode (110b) with a CHECK CONDITION status, a sense key of ILLEGAL REQUEST, and shall set the additional sense code to Command Sequence Error.

If the last WRITE BUFFER command of a set of one or more commands completes successfully, the microcode or control information shall be transferred to the control memory space of the logical unit. After a power-cycle or reset, the device shall revert to a vendor-specific condition. In this mode, the Data Out Buffer contains vendor-specific, self-describing microcode or control information.

Since the downloaded microcode or control information may be sent using several commands, when the logical unit detects the last download microcode with offsets and save mode WRITE BUFFER command has been received, the target shall perform any logical unit required verification of the complete set of downloaded microcode or control information prior to returning Good status for the last command. After the last command completes successfully the target shall generate a unit attention condition (see SAM-2) for all initiators except the one that issued the set of WRITE BUFFER commands. When reporting the unit attention condition, the target shall set the additional sense code to Microcode Has Been Changed.

If the complete set of WRITE BUFFER commands required to effect a microcode or control information change (one or more commands) are not received before a reset or power-on cycle occurs, the change shall not be effective and the new microcode or control information shall be discarded.

The Buffer ID field identifies a specific buffer within the logical unit. The vendor assigns buffer ID codes to buffers within the logical unit. A Buffer ID value of zero shall be supported. If more than one buffer is supported, additional buffer ID codes shall be assigned contiguously, beginning with one. If an unsupported buffer ID code is identified, the target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

The microcode or control information are written to the logical unit buffer starting at the location specified by the Buffer Offset field. The initiator shall send commands that conform to the offset boundary requirements (see Section 8.25.3). If the target is unable to accept the specified buffer offset, it shall return CHECK CONDITION status and it shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

The PARAMETER LIST LENGTH field specifies the maximum number of bytes that shall be present in the Data Out Buffer to be stored in the specified buffer beginning at the buffer offset. The initiator should attempt to ensure that the PARAMETER LIST LENGTH plus the buffer offset does not exceed the capacity of the specified buffer. (The capacity of the buffer may be determined by the Buffer Capacity field in the READ BUFFER descriptor.) If the Buffer Offset and PARAMETER LIST LENGTH fields specify a transfer in excess of the buffer capacity, the target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

8.64.7 Download microcode with offsets and save mode (0111b)

In this mode, the initiator may split the transfer of the vendor-specific microcode or control information over two or more WRITE BUFFER commands. If the logical unit is unable to accept this command because of some device condition, the target shall terminate each mode 111b WRITE BUFFER command with a CHECK CONDITION status, a sense key of ILLEGAL REQUEST, and shall set the additional sense code to Command Sequence Error.

If the last WRITE BUFFER command of a set of one or more commands completes successfully, the microcode or control information shall be saved in a non-volatile memory space (semiconductor, disk, or other). The saved downloaded microcode or control information shall then be effective after each power-cycle and reset until it is supplanted by another download microcode with save operation or download microcode with offsets and save operation. In this mode, the Data Out Buffer contains vendor-specific, self-describing microcode or control information.

Since the downloaded microcode or control information may be sent using several commands, when the logical unit detects the last download microcode with offsets and save mode WRITE BUFFER command has been received, the target shall perform any logical unit required verification of the complete set of downloaded microcode or control information prior to returning Good status for the last command. After the last command completes successfully the target shall generate a unit attention condition (see SAM-2) for all initiators except the one that issued the set of WRITE BUFFER commands. When reporting the unit attention condition, the target shall set the additional sense code to Microcode Has Been Changed.

If the complete set of WRITE BUFFER commands required to effect a microcode or control information change (one or more commands) are not received before a reset or power-on cycle occurs, the change shall not be effective and the new microcode or control information shall be discarded.

The Buffer ID field identifies a specific buffer within the logical unit. The vendor assigns buffer ID codes to buffers within the logical unit. A Buffer ID value of zero shall be supported. If more than one buffer is supported, additional buffer ID codes shall be assigned contiguously, beginning with one. If an unsupported buffer ID code is identified, the target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

The microcode or control information are written to the logical unit buffer starting at the location specified by the Buffer Offset field. The initiator shall conform to the offset boundary requirements. If the target is unable to accept the specified buffer offset, it shall return CHECK CONDITION status and it shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

The PARAMETER LIST LENGTH field specifies the maximum number of bytes that shall be present in the Data Out Buffer to be stored in the specified buffer beginning at the buffer offset. The initiator should attempt to ensure that the PARAMETER LIST LENGTH plus the buffer offset does not exceed the capacity of the specified buffer. (The capacity of the buffer may be determined by the Buffer Capacity field in the READ BUFFER descriptor.) If the Buffer Offset and PARAMETER LIST LENGTH fields specify a transfer in excess of the buffer capacity, the target shall return CHECK CONDITION status and shall set the sense key to ILLEGAL REQUEST with an additional sense code of Invalid Field In CDB.

8.64.8 Write data to echo buffer (1010b)

In this mode the target transfers data from the initiator and stores it in an echo buffer. An echo buffer is assigned in the same manner by the target as it would for a Write operation. Data shall be sent aligned on four-byte boundaries. The Buffer ID and Buffer Offset fields are ignored in this mode. Upon successful completion of a WRITE BUFFER command the data shall be preserved in the echo buffer unless there is an intervening command to any logical unit in which case it may be changed.

8.65 WRITE LONG command (3Fh)

The WRITE LONG command (Table 258) requests that the target write to the medium the data transferred by the initiator. The data passed during the WRITE LONG command shall include the data bytes and all ECC bytes to be written to the single logical block addressed in the command. The READ LONG command is usually issued before issuing a WRITE LONG command. The WRITE LONG data passed must be in the same order and must be the same number of bytes as the READ LONG command.

Table 258. WRITE LONG command (3Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	0	1	1	1	1	1	1
1	RESERVED [1]			0	0	0	0	RELADR [2]
2	(MSB)	LOGICAL BLOCK ADDRESS [3]						
:								
5								(LSB)
6	Reserved							
7	(MSB)	TRANSFER LENGTH [4]						
8								(LSB)
9	CONTROL [5]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the logical block at which the Write operation shall occur.

[3] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin if RELADR bit is zero (see note [4]).

[4] The BYTE TRANSFER LENGTH specifies the number of bytes of data that shall be transferred. A transfer length of zero indicates that no bytes shall be transferred. This condition shall not be considered as an error. The byte transfer length requested must be equal to the current block size plus all ECC bytes for a data transfer to occur. If an incorrect number of bytes is stated in the command block, this command terminates with a CHECK CONDITION status. The correct number of bytes can be determined from the information returned in the extended sense data bytes after issuing the REQUEST SENSE command. The REQUEST SENSE command shall result in the Illegal Field In CDB condition with the Illegal Request Sense key. The extended sense ILI bit shall be set. The extended sense information bytes contain the difference (residue) of the requested length minus the actual length in bytes. (Negative values are indicated by two's complement notation.) The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

8.66 WRITE SAME (10) command (41h)

The WRITE SAME (10) command (Table 259) requests that the target write the single block of data transferred by the initiator to the medium multiple times.

Table 259. WRITE SAME (10) command (41h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	0	0	0	0	0	1
1	RESERVED [1]			Reserved		PBDATA [2]	LBDATA [3]	RELADR [4]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [5]							
5								(LSB)
6	Reserved							
7	(MSB)							
	NUMBER OF BLOCKS [6]							
8								(LSB)
9	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Physical Block Data (PBDATA) bit of one requests that the target replace the first eight bytes of the data to be written to the current physical sector with the physical address of the sector currently being written using the physical sector format.

[3] A Logical Block Data (LBDATA) bit of one requests that the target replace the first four bytes of the data to be written to the current logical block with the LOGICAL BLOCK ADDRESS of the block currently being written.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first RELADR logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall occur. The most recent data written in the addressed logical block shall be returned.

[6] The NUMBER OF BLOCKS field specifies the number of contiguous logical blocks to be written. A NUMBER OF BLOCKS value of zero requests that all the remaining logical blocks on the medium be written.

[7] See Table 1, "CONTROL field," in Section 7.2.1.

8.67 WRITE SAME (16) command (93h)

The WRITE SAME (16) command (Table 260) requests that the target write the single block of data transferred by the initiator to the medium multiple times.

Table 260. WRITE SAME (16) command (93h)

Bit Byte	7	6	5	4	3	2	1	0
0	1	0	0	1	0	0	1	1
1	RESERVED [1]					PBDATA [2]	LBDATA [3]	RELADR [4]
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [5]							
9								(LSB)
10	(MSB)							
:	NUMBER OF BLOCKS [6]							
13								(LSB)
14	Reserved							
15	CONTROL [7]							

[1] Formerly LOGICAL UNIT NUMBER. See 7.1.1 for reservation requirements for this command.

[2] A Physical Block Data (PBDATA) bit of one requests that the target replace the first eight bytes of the data to be written to the current physical sector with the physical address of the sector currently being written using the physical sector format.

[3] A Logical Block Data (LBDATA) bit of one requests that the target replace the first four bytes of the data to be written to the current logical block with the LOGICAL BLOCK ADDRESS of the block currently being written.

[4] A Relative Address (RELADR) bit of one indicates that the LOGICAL BLOCK ADDRESS field is a two's complement displacement. This negative or positive displacement is to be added to the LOGICAL BLOCK ADDRESS last accessed on the logical unit to form the LOGICAL BLOCK ADDRESS for this command. This feature is only available when linking commands. The feature requires that a previous command in the linked group have accessed a block of data on the logical unit.

A RELADR bit of zero indicates that the LOGICAL BLOCK ADDRESS field specifies the first RELADR logical block of the range of logical blocks to be operated on by this command.

[5] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Read operation shall occur. The most recent data written in the addressed logical block shall be returned.

[6] The NUMBER OF BLOCKS field specifies the number of contiguous logical blocks to be written. A NUMBER OF BLOCKS value of zero requests that all the remaining logical blocks on the medium be written.

[7] See Table 1, “CONTROL field,” in Section 7.2.1.

8.68 Xor commands

The SCSI Xor commands are a special group of commands designed principally for use by array controllers in operating arrays of storage devices, though they are not limited to such a usage. These commands are members of command groups for direct access devices. Table 261 lists the Xor commands and the section number in which each is described. Mode page 10h “Xor control mode page” is used in conjunction with these commands and is listed in Section 8.13.12 (Table 168) with the other mode page descriptions.

Table 261. Xor commands defined in this section

Operation Code	Command Name	Section	Page
52h	XDREAD	8.69	390
50h	XDWRITE	8.71	393
51h	XPWRITE	8.75	399

Not all Seagate devices support the Xor commands. Where used by a particular model Seagate device, the individual drive’s Product Manual, Volume 1, for that model device specifies that the Xor commands are supported by the devices addressed by the Product Manual.

In storage arrays, an array controller organizes a group of storage devices into a redundancy group. Some areas within the address space of the storage array are used for check data. The check data is generated by performing a cumulative exclusive-or (Xor) operation with the data from other areas within the address space of the storage array known as protected data. This Xor operation can be performed by the array controller or by the storage device.

Performing the Xor operation in the storage device may result in a reduced number of data transfers across the interconnect. For example, when the Xor operation is done within the array controller four data transfer operations are needed for a typical update write sequence: a read transfer from the device containing protected data, a write transfer to the device containing protected data, a read transfer from the device containing check data, and a write transfer to the device containing check data. The array controller also does two internal Xor operations in this sequence. In contrast, during array controller supervised Xor operations, only three data transfer operations are needed: a write transfer to the device containing protected data, a read transfer from the device containing protected data, and a write transfer to the device containing check data. Note that the array controller doesn't do any internal Xor operations. In further contrast, during third party Xor operations, only two data transfer operations are needed: a write transfer from the array controller to the device containing protected data and a write transfer from the device containing protected data to the device containing check data. Note that the array controller only issues one command and does no Xor operations.

Performing the Xor operation in the device eliminates the need for the array controller to perform any Xor operations. An array controller performs three basic operations that require Xor functionality. These are the update write, regenerate, and rebuild operations.

For additional information on the use of the Xor commands see ANSI document X3T10/94-111r9.

8.69 XDREAD (10) command (52h)

The XDREAD (10) command (Table 262) requests that the target transfer to the initiator the Xor data generated by an XDWRITE or REGENERATE command.

Table 262. XDREAD (10) command (52h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	0	1	0	0	1	0
1	Reserved							
2 3 4 5	(MSB)	LOGICAL BLOCK ADDRESS [1]						(LSB)
6	Reserved							
7 8	(MSB)	TRANSFER LENGTH [1]						(LSB)
9	CONTROL [2]							

[1] The Xor data transferred is identified by LOGICAL BLOCK ADDRESS and Transfer Length that are the same as those specified in a prior XDWRITE or REGENERATE command. If a match is not found, the command is terminated with a CHECK CONDITION status. The sense data is set to ILLEGAL REQUEST: Invalid Field in CDB. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[2] See Table 1, “CONTROL field,” in Section 7.2.1.

8.70 XDREAD (32) command (7Fh)

The XDREAD (32) command (Table 263) requests that the target transfer to the initiator the Xor data generated by an XDWRITE or REGENERATE command. For additional information about Xor commands, see Section 8.68.

The Xor data transferred is identified by LOGICAL BLOCK ADDRESS and Transfer Length that are the same as those specified in a prior XDWRITE or REGENERATE command. If a match is not found, the command is terminated with a CHECK CONDITION status. The sense data is set to ILLEGAL REQUEST: Invalid Field in CDB.

Table 263. XDREAD (32) command (7Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	1	1	1	1	1	1
1	CONTROL [1]							
2	RESERVED [2]							
3	Reserved							
:								
6								
7	ADDITIONAL CDB LENGTH (18h) [3]							
8	(MSB)	SERVICE ACTION (0003h) [4]						
9								(LSB)
10	Reserved							
11								
12	(MSB)	LOGICAL BLOCK ADDRESS [5]						
:								
19								(LSB)
20	Reserved							
:								
27								
28	(MSB)	TRANSFER LENGTH [6]						
:								
31								(LSB)

[1] See Table 1, “CONTROL field,” in Section 7.2.1.

[2] See 7.1.1 for reservation requirements for this command.

[3] The Additional CDB Length field contains the length in 4-byte words of the Additional CDB field.

[4] 0003h is the Service Action Code for the XDREAD (32) command.

[5] The LOGICAL BLOCK ADDRESS field specifies the starting LOGICAL BLOCK ADDRESS at which the target shall write the Xor result data on its own medium.

[6] The TRANSFER LENGTH field specifies the number of logical blocks that shall be transferred to the XDWRITE target for the Xor operation. It also specifies the number of blocks to be written to the medium after the Xor operation. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

8.71 XDWRITE (10) command (50h)

The XDWRITE (10) command (see Table 264) requests that the target Xor the data transferred to it with the data on the medium. The resulting Xor data is stored in the target's buffer. The disposition of the data transferred from the initiator is controlled by the disable write bit.

The resulting Xor data is retained in the target's buffer until it is retrieved by an XDREAD (10) command with starting logical block address and transfer length fields that match the starting logical block address and transfer length of this command. For additional information about Xor commands, see Section 8.68.

Table 264. XDWRITE (10) command (50h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	0	1	0	0	0	0
1	Reserved			DPO [1]	FUA [2]	DISABLE WRITE [3]	Reserved	
2	(MSB)	LOGICAL BLOCK ADDRESS [4]						
:								
5								(LSB)
6	Reserved							
7	(MSB)	TRANSFER LENGTH [5]						
8								(LSB)
9	CONTROL [6]							

[1] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (caching enabled).

[2] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[3] A DISABLE WRITE bit of zero indicates that the data transferred from the initiator shall be written to the medium after the Xor operation is complete. A DISABLE WRITE bit of one indicates that the data shall not be written to the medium.

[4] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin.

[5] The TRANSFER LENGTH field specifies the number of logical blocks that shall be transferred to the XDWRITE target for the Xor operation. It also specifies the number of blocks to be written to the medium after the Xor operation. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

8.72 XDWRITE (32) command (7Fh)

The XDWRITE (32) command (see Table 265) requests that the target Xor the data transferred to it with the data on the medium. The resulting Xor data is stored in the target's buffer. The disposition of the data transferred from the initiator is controlled by the disable write bit.

The resulting Xor data is retained in the target's buffer until it is retrieved by an XDREAD (32) command with starting logical block address and transfer length fields that match the starting logical block address and transfer length of this command. For additional information about Xor commands, see Section 8.68.

Table 265. XDWRITE (32) command (7Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	1	1	1	1	1	1
1	CONTROL [1]							
2	RESERVED [2]							
3	Reserved							
6								
7	ADDITIONAL CDB LENGTH (18h) [3]							
8	(MSB)	SERVICE ACTION (0004h) [4]						
9								(LSB)
10	Reserved			DPO [5]	FUA [6]	DISABLE WRITE [7]	RESERVED [2]	
11	Reserved							
12	(MSB)	LOGICAL BLOCK ADDRESS [8]						
:								
19								(LSB)
20	Reserved							
:								
27								
28	(MSB)	TRANSFER LENGTH [9]						
:								
31								(LSB)

[1] See Table 1, “CONTROL field,” in Section 7.2.1.

[2] See 7.1.1 for reservation requirements for this command.

[3] The Additional CDB Length field contains the length in 4-byte words of the Additional CDB field.

[4] 0004h is the Service Action Code for the XDWRITE (32) command.

[5] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (caching enabled).

[6] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[7] A DISABLE WRITE bit of zero indicates that the data transferred from the initiator shall be written to the medium after the Xor operation is complete. A DISABLE WRITE bit of one indicates that the data shall not be written to the medium.

[8] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin.

[9] The TRANSFER LENGTH field specifies the number of logical blocks that shall be transferred to the XDWRITE target for the Xor operation. It also specifies the number of blocks to be written to the medium after the Xor operation. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

8.73 XDWRITEREAD (10) command (53h)

The XDWRITEREAD (10) command (see Table 266) requests that the target XOR the data transferred (data-out) with the data on the medium and return the resulting XOR data (data-in). This is the equivalent to an XDWRITE (10) followed by an XDREAD (10) with the same Logical Block Address and Transfer Length. This command is only available on transport protocols supporting bidirectional commands. For additional information about Xor commands, see Section 8.68.

Table 266. XDWRITEREAD (10) command (53h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	0	1	0	0	1	1
1	Reserved			DPO [1]	FUA [2]	DISABLE WRITE [3]	Reserved	
2	(MSB)							
:	LOGICAL BLOCK ADDRESS [4]							
5								(LSB)
6	Reserved							
7	(MSB)							
	TRANSFER LENGTH [5]							
8								(LSB)
9	CONTROL [6]							

[1] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (caching enabled).

[2] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[3] A DISABLE WRITE bit of zero indicates that the data transferred from the initiator shall be written to the medium after the Xor operation is complete. A DISABLE WRITE bit of one indicates that the data shall not be written to the medium.

[4] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin.

[5] The TRANSFER LENGTH field specifies the number of logical blocks that shall be transferred to the XDWRITE target for the Xor operation. It also specifies the number of blocks to be written to the medium after the Xor operation. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[6] See Table 1, “CONTROL field,” in Section 7.2.1.

8.74 XDWRITEREAD (32) command (7Fh)

The XDWRITEREAD (32) command (Table 267) requests that the target XOR the data transferred (data-out) with the data on the medium and return the resulting XOR data (data-in). This is the equivalent to an XDWRITE (32) followed by an XDREAD (32) with the same Logical Block Address and Transfer Length. This command is only available on transport protocols supporting bidirectional commands. For additional information about Xor commands, see Section 8.68.

Table 267. XDWRITEREAD (32) command (7Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	1	1	1	1	1	1
1	CONTROL [1]							
2	RESERVED [2]							
3	Reserved							
5								
6								
7	ADDITIONAL CDB LENGTH (18h) [3]							
8	(MSB)	SERVICE ACTION (0007h) [4]						
9								(LSB)
10	Reserved			DPO [5]	FUA [6]	DISABLE WRITE [7]	Reserved	
11	Reserved							
12	(MSB)	LOGICAL BLOCK ADDRESS [8]						
:								
19								(LSB)
20	Reserved							
:								
27								
28	(MSB)	TRANSFER LENGTH [9]						
:								
31								(LSB)

[1] See Table 1, “CONTROL field,” in Section 7.2.1.

[2] See 7.1.1 for reservation requirements for this command.

[3] The Additional CDB Length field contains the length in 4-byte words of the Additional CDB field.

[4] 0007h is the Service Action Code for the XDWRITEREAD (32) command.

[5] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (caching enabled).

[6] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[7] A DISABLE WRITE bit of zero indicates that the data transferred from the initiator shall be written to the medium after the Xor operation is complete. A DISABLE WRITE bit of one indicates that the data shall not be written to the medium.

[8] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin.

[9] The TRANSFER LENGTH field specifies the number of logical blocks that shall be transferred to the XDWRITE target for the Xor operation. It also specifies the number of blocks to be written to the medium after the Xor operation. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

8.75 XPWRITE (10) command (51h)

The XPWRITE (10) command (see Table 268) requests that the target Xor the data transferred to it with the data on the medium and then writes the Xor data to the medium. For additional information about Xor commands, see Section 8.68.

Table 268. XPWRITE (10) command (51h)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	0	1	0	0	0	1
1	Reserved			DPO [1]	FUA [2]	Reserved		
2	(MSB)							
3	LOGICAL BLOCK ADDRESS [3]							
4								
5								(LSB)
6	Reserved							
7	(MSB)							
8	TRANSFER LENGTH [4]							(LSB)
9	CONTROL [5]							

[1] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (caching enabled).

[2] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[3] The LOGICAL BLOCK ADDRESS field specifies the starting LOGICAL BLOCK ADDRESS at which the target shall read data from its medium. It also specifies the starting LOGICAL BLOCK ADDRESS at which the Xor result data is to be written to the medium.

[4] The TRANSFER LENGTH field specifies the number of blocks to be read from the XPWRITE target medium for the Xor operation. It also specifies the number of blocks to be written to the medium after the Xor operation. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

[5] See Table 1, “CONTROL field,” in Section 7.2.1.

8.76 XPWRITE (32) command (7Fh)

The XPWRITE (32) command (Table 269) requests that the target XOR the data transferred with the data on the medium and then write the XOR data to the medium.

The resulting Xor data is retained in the target's buffer until it is retrieved by an XDREAD (32) command with starting logical block address and transfer length fields that match the starting logical block address and transfer length of this command. For additional information about Xor commands, see Section 8.68.

Table 269. XPWRITE (32) command (7Fh)

Bit Byte	7	6	5	4	3	2	1	0
0	0	1	1	1	1	1	1	1
1	CONTROL [1]							
2	RESERVED [2]							
3	Reserved							
:								
6								
7	ADDITIONAL CDB LENGTH (18h) [3]							
8	(MSB)	SERVICE ACTION (0006h) [4]						
9								(LSB)
10	Reserved			DPO [5]	FUA [6]	Reserved		
11	Reserved							
12	(MSB)							
:	LOGICAL BLOCK ADDRESS [7]							
19								(LSB)
20	Reserved							
:								
27								
28	(MSB)							
:	TRANSFER LENGTH [8]							
31								(LSB)

[1] See Table 1, “CONTROL field,” in Section 7.2.1.

[2] See 7.1.1 for reservation requirements for this command.

[3] The Additional CDB Length field contains the length in 4-byte words of the Additional CDB field.

[4] 0006h is the Service Action Code for the XPWRITE (32) command.

[5] If the Disable Page Out (DPO) bit is set to one, no data is cached. The DPO bit is only meaningful if the RCD bit of MODE SELECT Page 8 is set false (caching enabled).

[6] A Force Unit Access (FUA) bit of one indicates that the WRITE command shall not return Good status until the logical blocks have actually been written on the media. The FUA bit is only meaningful if the WCE bit of MODE SENSE page 8 is true.

[7] The LOGICAL BLOCK ADDRESS specifies the logical block at which the Write operation shall begin.

[8] The TRANSFER LENGTH field specifies the number of logical blocks that shall be transferred to the XDWRITE target for the Xor operation. It also specifies the number of blocks to be written to the medium after the Xor operation. The TRANSFER LENGTH field is constrained by the MAXIMUM TRANSFER LENGTH field in the Block Limits VPD page (see Section 8.13.13.2).

9.0 Drive features

9.1 S.M.A.R.T. system

Some drive families mentioned in Section 1.1 implement what is called in the industry the S.M.A.R.T. system. S.M.A.R.T. is an acronym for Self-Monitoring Analysis and Reporting Technology. The intent of the S.M.A.R.T. system is to recognize conditions that indicate imminent drive failure and provide sufficient warning to the host system of impending failure. The host system may use the information provided to trigger it to perform diagnostic, preventative, and/or protective functions (e.g., data backup).

The initiator sets up the parameters for S.M.A.R.T. operation using MODE SELECT Informational Exceptions Control page 1Ch. The drive reports information about S.M.A.R.T. operation using REQUEST SENSE Additional Sense Code 5D 00 and MODE SENSE data page 1Ch. Refer to sections 8.11, 8.12, 8.13, and 8.14 for the description of the MODE SELECT/MODE SENSE commands and Section 8.13.15 for more details on the Informational Exceptions Control page. Refer to the individual drive's Product Manual, Volume 1, to determine the extent of its implementation of the S.M.A.R.T. system.

9.2 Self-test operations

9.2.1 Default self-test

The default self-test is mandatory for all device types that support the SEND DIAGNOSTICs command. The response is simply a GOOD status if the test is successful or a CHECK CONDITION status if the test fails.

9.2.2 The short and extended self-tests

There are two optional types of self-test that may be invoked using the Self-Test Code field in the SEND DIAGNOSTICs command: a short self-test and an extended self-test. The goal of the short self-test is to quickly identify if the logical unit is faulty. A goal of the extended self-test routine is to simplify factory testing during integration by having logical units perform more comprehensive testing without initiator intervention. A second goal of the extended self-test is to provide a more comprehensive test to validate the results of a short self-test, if its results are judged by the initiator to be inconclusive.

The criteria for the short self-test are that it has one or more segments and completes in two minutes or less. The criteria for the extended self-test are that it is has one or more segments and that the completion time is vendor-specific. Any tests performed in the segments are vendor-specific.

The following are examples of segments:

a. An electrical segment wherein the logical unit tests its own electronics. The tests in this segment are vendor-specific, but some examples of tests that might be included are: a read/write circuitry test and/or a test of the read/write head elements;

b. A seek/servo segment wherein a device tests it capability to find and servo on data tracks; and

c. A read/verify scan segment wherein a device performs read scanning of some or all of the medium surface.

The tests performed in the segments may be the same for the short and extended self-tests. The time required by a logical unit to complete its extended self-test is reported in the Extended Self-Test Completion Time field in the Control mode page (see Section 8.13.9).

9.2.3 Self-test modes

There are two modes for short and extended self-tests: a foreground mode and a background mode. These modes are described in the following sections.

9.2.3.1 Foreground mode

When a target receives a SEND DIAGNOSTICs command specifying a self-test to be performed in the foreground mode, the target shall return status for that command after the self-test has been completed. Not all Seagate drives support this mode.

While performing a self-test in the foreground mode, the target shall respond to all commands except INQUIRY, REPORT LUNS, and REQUEST SENSE with a CHECK CONDITION status, a sense key of Not Ready and an additional sense code of Logical Unit Not Ready, Self-Test In Progress.

If a target is performing a self-test in the foreground mode and a test segment error occurs during the test, the target shall update the Self-test results log page (see Section 8.10.11) and report CHECK CONDITION status with a sense key of Hardware Error and an additional sense code of Logical Unit Failed Self-Test. The initiator may obtain additional information about the failure by reading the Self-test results log page. If the target is unable to update the Self-test results log page it shall return a CHECK CONDITION status with a sense key of Hardware Error and an additional sense code of Logical Unit Unable To Update Self-Test Log.

An initiator should reserve the logical unit before initiating a self-test in the foreground mode. An initiator may terminate a self-test that is being performed in the foreground mode using an Abort Task, ABORT TASK SET, or Clear Task Set task management function. If a task manager receives an Abort Task, ABORT TASK SET, or Clear Task Set task management function while performing a self-test in the foreground mode, the it shall abort the self-test and update the Self-test results log page (see Section 8.10.11).

9.2.3.2 Background mode

When a target receives a SEND DIAGNOSTICs command specifying a self-test to be performed in the background mode, the target shall return status for that command as soon as the command descriptor block has been validated.

After returning status for the SEND DIAGNOSTICs command specifying a self-test to be performed in the background mode, the target shall initialize the Self-test results log page (see Section 8.10.11) as follows. The self-test code from the SEND DIAGNOSTICs command shall be placed in the Self-Test Code field in the log page. The Self-Test Results field shall be set to Fh. After the Self-test results log page is initialized, the target shall begin the first self-test segment.

While the target is performing a self-test in the background mode, it shall terminate with a CHECK CONDITION status any SEND DIAGNOSTICs command it receives that meets one of the following criteria:

a. The Self-Test bit is one; or

b. The Self-Test Code field contains a value other than 000b or 100b.

When terminating the SEND DIAGNOSTICs command, the sense key shall be set to Not Ready and the additional sense code shall be set to Logical Unit Not Ready, Self-Test In Progress.

While performing a self-test in the background mode, the target shall suspend the self-test to service any other commands received with the exceptions listed in Table 270. Suspension of the self-test to service the command shall occur as soon as possible, but shall never take longer than two seconds.

Table 270: Exception commands for background self-tests

Device type [1]	Command	Referenc e
All device types	SEND DIAGNOSTICs (with Self-Test Code field set to 100b)	8.47
	WRITE BUFFER (with the mode set to any download microcode option	8.64
Direct access	FORMAT UNIT	SBC
	START/STOP UNIT (stop only)	

[1] Device types not listed in this table do not have commands that are exceptions for background self-tests, other than those listed above for all device types.

If one of the exception commands listed in Table 270 is received, the target shall abort the self-test, update the self-test log, and service the command as soon as possible but not longer than two seconds after the command descriptor block has been validated.

Note. An initiator may terminate a self-test that is being performed in the background mode by issuing a SEND DIAGNOSTICs command with the Self-Test Code field set to 100b (Abort background self-test function).

9.2.3.3 Elements common to foreground and background self-test modes

The Progress Indication field returned in response to a REQUEST SENSE command (see Section 8.38) may be used by the initiator at any time during execution of a self-test to poll the logical unit's progress. While executing a self-test unless an error has occurred, a target shall respond to a REQUEST SENSE command by returning a sense key of Not Ready and an additional sense code of Logical Unit Not Ready, Self-Test In Progress with the sense key specific bytes set for progress indication.

The initiator may obtain information about the twenty most recently completed self-tests by reading the Self-test results log page (see Section 8.10.11). This is the only method for an initiator to obtain information about self-tests performed in the background mode.

Table 271 summarizes when a logical unit returns status after receipt of a self-test command, how an initiator may abort a self-test, how a logical unit handles new commands that are received while a self-test is in progress, and how a logical unit reports a self-test failure.

Table 271: Self-test mode summary

Mode	When status is returned	How to abort the self-test	Processing of subsequent commands while self-test is executing	Self-test failure reporting
Fore-ground	After the self-test is complete	Abort Task task management function	If the command is INQUIRY, REPORT LUNS, or REQUEST SENSE, process normally. Otherwise, terminate with CHECK CONDITION status, Not Ready sense key, and Logical Unit Not Ready, Self-Test In Progress additional sense code.	Terminate with CHECK CONDITION status, Hardware Error sense key, and Logical Unit Failed Self-Test or Logical Unit Unable To Update Self-Test Log additional sense code.
Back-ground	After the CDB is validated	SEND DIAGNOSTICs command with Self-Test Code field set to 100b	Process the command, except as described in 9.2.3.2.	initiator checks Self-test results log page (see 8.10.11) after the Progress Indication field returned from REQUEST SENSE indicates the self-test is complete.

9.3 Alternate error detection for the asynchronous information phases (AIP)—Command, Message, and Status

9.3.1 Error detection for asynchronous information phases

This section 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 can 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.

9.3.2 Protection code

The following are the covered signals to be encoded and details of the protection code to be used on the asynchronous information phases.

9.3.2.1 Covered signals

Table 272 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 272.

Table 272: Signals to be covered by the protection code and their bit locations

Code word 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 [1]
9	DB(9)	Reserved [1]
10	RSVD	Reserved [2]
11	RSVD	Reserved [2]
12	RSVD	Reserved [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

[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 could 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.

9.3.2.2 Code description

The protection code (see Table 273) is a cyclic binary BCH code.

Table 273: 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 21. 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.

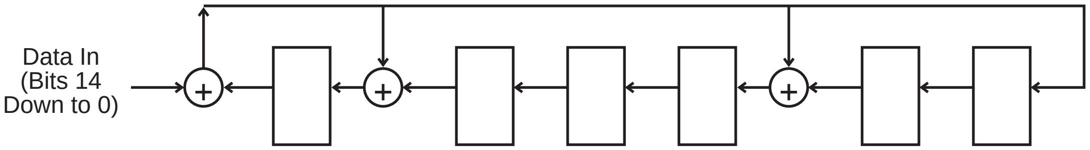


Figure 21. Protection code generator

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.

9.3.2.3 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 possible errors.

9.3.3 Protection code usage

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 SCSI target port 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.

9.3.3.1 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.

9.3.3.2 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 possible times when a SCSI device could 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 SCSI target port 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.

9.3.3.3 Disabling protection code checking

The removal and insertion of a SCSI device could 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 possible ways that a SCSI device could 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).

9.3.4 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).

9.3.5 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 clause 11.1 of SPI-4.

Some examples of how API works are given in SPI-4 Annex M, subsection M.6.

9.4 Removal and insertion of SCSI devices (popularly known as “hot plugging”)

9.4.1 Removal and insertion of SCSI devices overview

This section 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.

9.4.2 Case 1—Power off during removal or insertion

a. All SCSI devices are powered off during physical reconfiguration.

9.4.3 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. The translation of the 1 ms time to mechanical provisions is vendor-specific.

9.4.4 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 ANSI SPI-4 specification, T10/1365D, sections 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. 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 mF. For single-ended applications, SCSI bus terminations shall use voltage regulation.

Note. In a multimode environment any insertion or removal that changes the SCSI bus mode causes a transceiver mode change reset event (see Section 5.4.4).

9.4.5 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 ANSI SPI-4 specification, T10/1365D, sections 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. 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 mF. For single-ended applications, SCSI bus terminations shall use voltage regulation.

Note. In a multimode environment, any insertion or removal that changes the bus mode causes a transceiver mode change reset event (see Section 5.4.4).

Note. LVD SCSI devices may require more stringent system design to tolerate transients that occur during Case 4 insertion or removal.

9.5 SPI-3 to SCSI-2 terminology mapping

This section contains a mapping of terminology used in SCSI-2 to the terminology used in this manual (see Table 274).

Table 274: 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

10.0 Seagate Technology support services

Online Services

Internet

www.seagate.com for information about Seagate products and services. Worldwide support is available 24 hours daily by e-mail for your questions.

Presales Support: www.seagate.com/support/email/email_presales.html *or* DiscPresales@Seagate.com
Technical Support: www.seagate.com/support/email/email_disc_support.html *or* DiscSupport@Seagate.com

mySeagate

my.seagate.com is the industry's first Web portal designed specifically for OEMs and distributors. It provides self-service access to critical applications, personalized content and the tools that allow our partners to manage their Seagate account functions. Submit pricing requests, orders and returns through a single, password-protected Web interface—anytime, anywhere in the world.

For support, questions and comments:
E-mail: partner.support@seagate.com
Phone (direct): +1-405-324-4740

Phone (toll-free): 1-877-347-2444 (US and Canada), or access a complete list of region-specific international toll-free phone numbers at my.seagate.com

reseller.seagate.com

reseller.seagate.com supports Seagate resellers with product information, program benefits and sales tools. You may register for customized communications that are not available on the web. These communications contain product launch, EOL, pricing, promotions and other channel-related information. To learn more about the benefits or to register, go to reseller.seagate.com, any time, from anywhere in the world.

reseller.seagate.com partner support
24 hours a day, seven days a week:
By web form: reseller.seagate.com/contact/forms
By phone: Access a complete list of regional phone numbers at reseller.seagate.com/support/

Seagate Online Store

US customers can purchase Seagate disc drives 24 hours daily at www.seagate.com/buy/online/.

Automated Phone Services

SeaFONE® (1-800-SEAGATE) is the Seagate toll-free number (1-800-732-4283) to access our automated self-help and directory assistance for Seagate support services. Using a touch-tone phone, you can find service and support phone numbers, answers to commonly asked questions, troubleshooting tips and specifications for disc drives 24 hours daily. International callers can reach this service by dialing +1-405-324-4770.

Presales, Technical Support and Customer Service

Presales Support

Our Presales Support staff can help you determine which Seagate products are best suited for your specific application or computer system.

Technical Support

If you need help installing your drive, consult your system's documentation or contact the dealer's support services department for assistance specific to your system. Seagate technical support is also available to assist you online at support.seagate.com or through one of our technical support service centers. Have your system configuration information and your drive's "ST" model number available.

SeaTDD™ (+1-405-324-3655) is a telecommunications device for the deaf (TDD). You can send questions or comments 24 hours daily and exchange messages with a technical support specialist during normal business hours for the technical support call center in your region.

Warranty Service

Seagate offers worldwide customer support for Seagate drives. Seagate distributors, OEMs and other direct customers should contact their Seagate service center representative for warranty-related issues. Resellers or end users of drive products should contact their place of purchase or one of the Seagate warranty service centers for assistance. Have your drive's "ST" model number and serial number available.

Authorized Service Centers

In some countries outside the US, you can contact an Authorized Service Center for service.

USA/Canada/Latin America Support Services

Presales Support

Call Center	Toll-free	Direct dial	FAX
Americas	1-877-271-3285[1]	+1-405-324-4730[1]	+1-405-324-4704

Technical Support

Call Center	Toll-free	Direct dial	FAX
Americas	1-800-SEAGATE[2]	+1-405-324-4700[3]	+1-405-324-4702

Warranty Service

Call Center	Toll-free	Direct dial	FAX / Internet
USA, Mexico and Latin America	1-800-468-3472[4]	+1-405-324-4720[4]	+1-405-324-4722
Canada			
Memofix[5]	1-800-636-6349	+1-905-660-4936	+1-905-660-4951 www.memofix.com
Brazil			
MA Centro de Serviços[5]	—	+55-21-2509-7267	+55-21-2507-6672 www.mainformatica.com.br/produtos.htm

[1] Hours of operation are 8:00 A.M. to 11:45 A.M. and 1:00 P.M. to 6:00 P.M., Monday through Friday (Central time)
[2] For product-specific phone number
[3] Hours of operation are 8:00 A.M. to 8:00 P.M., Monday through Friday (Central time)
[4] Hours of operation are 8:30 A.M. to 12:15 P.M. and 1:30 P.M. to 5:30 P.M., Monday through Friday (Central time)
[5] Authorized Service Center

European Support Services

For presales and technical support in Europe, dial the toll-free number for your specific country. If your country is not listed here, dial our presales and technical support call center at +1-405-324-4714 from 8:00 A.M. to 11:45 A.M. and 1:00 P.M. to 5:00 P.M. (Central Europe time) Monday through Friday. The presales and technical support call center is located in Oklahoma City, USA.

For European warranty service, dial the toll-free number for your specific country. If your country is not listed here, dial our European call center at +31-20-316-7222 from 8:30 A.M. to 5:00 P.M. (Central Europe time) Monday through Friday. The warranty service call center is located in Amsterdam, The Netherlands.

Toll-Free Support Numbers

Call Center	**Presales and Technical Support**	**Warranty Service**
Austria	0 800-20 12 90	0 800-20 12 90
Belgium	00 800-47324283 (00 800-4SEAGATE)	00 800-47324289
Denmark	00 800-47324283	00 800-47324289
France	00 800-47324283	00 800-47324289
Germany	00 800-47324283	00 800-47324289
Ireland	00 800-47324283	00 800-47324289
Italy	00 800-47324283	00 800-47324289
Netherlands	00 800-47324283	00 800-47324289
Norway	00 800-47324283	00 800-47324289
Poland	00 800-311 12 38	00 800-311 12 38
Spain	00 800-47324283	00 800-47324289
Sweden	00 800-47324283	00 800-47324289
Switzerland	00 800-47324283	00 800-47324289
Turkey	00 800-31 92 91 40	00 800-31 92 91 40
United Kingdom	00 800-47324283	00 800-47324289

FAX Services—All European Countries (toll call)

Warranty Service	+31-20-653-3513

Africa/Middle East Support Services

For presales and technical support in Africa and the Middle East, dial our presales and technical support call center at +1-405-324-4714 from 8:00 A.M. to 11:45 A.M. and 1:00 P.M. to 5:00 P.M. (Central Europe time) Monday through Friday. The presales and technical support call center is located in Oklahoma City, USA.

For warranty service in Africa and the Middle East, dial our European call center at +31-20-316-7222 from 8:30 A.M. to 5:00 P.M. (Central Europe time) Monday through Friday, or send a FAX to +31-20-653-3513. The warranty service center is located in Amsterdam, The Netherlands.

Asia/Pacific Support Services

For Asia/Pacific presales and technical support, dial the toll-free number for your specific country. The Asia/ Pacific toll-free numbers are available from 6:00 A.M. to 10:45 A.M. and 12:00 P.M. to 6:00 P.M. (Australian Eastern time) Monday through Friday, except as noted. If your country is not listed here, direct dial one of our technical support locations. Warranty service is available from 9:00 A.M. to 6:00 P.M. April through October, and 10:00 P.M. to 7:00 P.M. November through March (Australian Eastern time) Monday through Friday, except as noted.

Call Center	**Toll-free**	**Direct dial**	**FAX**
Australia	1800-14-7201	—	—
China (Chinese)[1, 4]	800-810-9668	+86-10-6225-5336	—
Hong Kong	800-90-0474	—	—
Hong Kong (Chinese)[1, 4]	001-800-0830-1730	—	—
India[2, 4]	1-600-33-1104	—	—
Indonesia	001-803-1-003-2165	—	—
Japan[3]	0034 800 400 554	—	—
Malaysia	1-800-80-2335	—	—
New Zealand	0800-443988	—	—
Singapore	800-1101-150	—	+65-6488-7525
Taiwan (Chinese)[1, 4]	00-800-0830-1730	—	—
Thailand	001-800-11-0032165	—	—

Warranty Service

Call Center	**Toll-free**	**Direct dial**	**FAX**
Asia/Pacific	—	+65-6485-3595	+65-6485-4860
Australia	1800-12-9277	—	—
India[4]	—	+91-44-821-6164	+91-44-827-2461

[1] Hours of operation are 8:30 A.M. to 5:30 P.M., Monday through Friday (Australian Western time).
[2] Hours of operation are 9:00 A.M. to 6:00 P.M., Monday through Saturday.
[3] Hours of operation are 9:30 A.M. to 6:30 P.M., Monday through Friday (Japan time).
[4] Authorized Service Center.

Index

Seagate Technology LLC
920 Disc Drive, Scotts Valley, California 95066-4544, USA
Publication Number: 75789509, Rev. C, Printed in USA