INTERNATIONAL STANDARD

ISO/IEC 9318-2

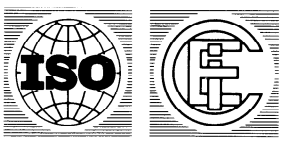
First edition
1990-12-15

Information technology — Intelligent Peripheral Interface

Part 2:
Device specific command set for magnetic disk drives

Technologies de l'information — Interface pour les périphériques intelligents —

Partie 2: Jeu de commandes spécifiques appareil pour les unités disques magnétiques



Reference number
ISO/IEC 9318-2 : 1990 (E)

Contents

Figures

Tables

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

International Standard ISO/IEC 9318-2 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

ISO/IEC 9318 consists of the following parts, under the general title *Information technology — Intelligent Peripheral Interface*:

- *Part 1: Physical level*
- *Part 2: Device specific command set for magnetic disk drives*
- *Part 3: Device generic command set for magnetic and optical disk drives*
- *Part 4: Device generic command set for magnetic tape drives*

Annex A forms an integral part of this part of ISO/IEC 9318. Annexes B, C and D are for information only.

Introduction

This part of ISO/IEC 9318 does not replace any existing standard, but it does complement other Intelligent Peripheral Interface (IPI) standards (see clause 2).

This part of ISO/IEC 9318 provides a definition of the device specific interface portion of a series of standards called the Intelligent Peripheral Interface (IPI) ISO/IEC 9318, a high performance, general-purpose parallel peripheral interface. This part of ISO/IEC 9318, responds to an industry market need (expressed both by users and manufacturers) to limit the increasing costs in hosts associated with changes in peripherals.

This page intentionally left blank

Information technology — Intelligent Peripheral Interface

Part 2:
Device specific command set for magnetic disk drives

1 Scope

This part of ISO/IEC 9318 describes the Logical Level 2 (device level) Interface for disk drives. See clause 6 of ISO/IEC 9318-1 for explanation of the levels.

The physical, electrical, and configuration characteristics and the transmission protocol of this interface are in accordance with ISO/IEC 9318-1. The interface is capable of handling data rates from 0 to at least 10 Moctets/s, depending on driver and receiver classes.

The purpose of this part of ISO/IEC 9318 is to facilitate the development and utilization of a device level interface which permits the interconnection of disk slave peripherals to a controller.

This part of ISO/IEC 9318 does not replace any existing standard, but it does complement other Intelligent Peripheral Interface (IPI) standards (see clause 2).

This part of ISO/IEC 9318 provides a definition of the device specific portion of a family of standards called the Intelligent Peripheral Interface (IPI), a high performance, general-purpose parallel peripheral interface.

The intent of the IPI is to isolate the host (CPU), both hardware and software, from changes in peripherals by providing a "function generic" command set to allow the connection of multiple types of peripherals (disks, printers, tapes, communications). To smooth the transition from the current methods to the generic approach, the IPI supports device-specific command sets, such as this one, to aid in bridging the gap between the two approaches.

To accomplish this set of goals, the design of the IPI includes device-specific and device-generic command sets, both utilizing a common physical bus. The device-specific command set provides:

- Device-oriented control;
- Physical Data Addressing;
- Timing-Critical Operations;
- Lower Device Cost.

The device-generic command set provides a higher level of functionality and portability. It includes:

- Host/Device Independence;
- Logical Data Addressing;
- Timing Independence;
- Command Queuing Capability.

A system is not restricted to the use of one level of command set or the other. It is possible that both levels of command sets will be utilized with a given system's architecture to balance such parameters as system performance, cost, and

peripheral availability. It is also possible for the host to provide for migration from device-specific to device-generic levels while still retaining the same physical interface.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 9318. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 9318 are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 646:1983, *Information processing - ISO 7-bit coded character set for information interchange.*

ISO/IEC 9318-1:—[1)] *Information technology - Intelligent Peripheral Interface Part 1: Physical Level*

1) To be published.

3 Definitions and conventions

3.1 Definitions

For the purpose of this part of ISO/IEC 9318, the following definitions apply:

3.1.1 Bus Control: The physical-level octet placed on BUS A by the master during the Bus Control sequence. It is used to define the bus configuration and information type for the subsequent Information Transfer. The three Bus Control types are: Command Control, Response Control, and Data Control.

3.1.2 Bus Exchange: The Bus Control sequence (initiated by the master) and the Ending Status sequence (initiated by the slave), which are used to frame an Information Transfer.

3.1.3 Busy: Busy indicates that the slave is unable to respond to the master's request and may be reported at either the Physical Interface (Selection Sequence) or at the Logical Interface (Ending Status Sequence).

3.1.4 Command: The information transferred from the master to the slave following a Command Control. The information contains command-specific parameters.

3.1.5 Command Control: A Bus Control that specifies that command information is to be transferred from the master to the slave during the Information Transfer and identifies the command type.

3.1.6 Data Control: A Bus Control that specifies that data information is to be transferred during the Information Transfer, and specifies various data transfer requirements, including the orientation of the data on the disk.

3.1.7 Defect Map: A list of the defects recorded by the drive manufacturer on the slave.

3.1.8 Format Specification: An ordered list of parameters that specify the format of the tracks and sectors on the disk.

3.1.9 Information Transfer: The Physical Interface transfer of commands, responses, and data that are framed by a Bus Exchange.

3.1.10 Level 2: A device-level-protocol interface in which the master is aware of the specific characteristics of the device under its control.

3.1.11 Level 3: An intelligent interface oriented to the generic characteristics of devices, but not the characteristics unique to the device.

3.1.12 Logical Interface: All operations above the Physical Interface.

3.1.13 Master Status: The status placed on BUS A by the master during the Ending Status sequence after the Information Transfer.

3.1.14 parameter: The information octets transferred after a Command or Response Bus Control. Command parameters further define a command or contain specific information about a command, such as head number, cylinder number, etc. Response parameters contain information on the state of the slave, such as status, current head, current cylinder, etc.

3.1.15 Physical Interface: The mechanical, electrical, and bus protocol characteristics specified in ISO/IEC 9318.

3.1.16 physical sector: This term refers to the sectors identified by the disk drive, relative to index, with zero as the first.

3.1.17 response: The information transferred from the slave to the master following a Response Control. The information contains parameters specific to the response type.

3.1.18 Response Control: A Bus Control which specifies that information about a response type is to be transferred from the slave to the master during the Information Transfer and identifies the type of response.

3.1.19 RPS: This is an abbreviation for Rotational Position Sensing, a means for the slave to notify the master of the relative head position with respect to index.

3.1.20 RPS Pulse: The time defined by the master on Load Position and Load RPS Target Sector Address commands during which the Class 2 RPS Interrupt shall be asserted.

3.1.21 Slave Status: The status placed on the BUS B by the slave during the Ending Status sequence after an Information Transfer.

3.1.22 status response: The status supplied to the master when executing a Read Status Response that contains the exception status bits.

3.1.23 Target Sector: The physical sector designated by the master during the Load Position command or Load RPS Target Sector Address command using the RPS Target Sector Address parameter.

3.2 Conventions

Certain terms used herein are the proper names of signals. These are printed in uppercase to avoid possible confusion with other uses of the same words (e.g., ATTENTION IN). Any lowercase uses of these words have the normal American-English meaning.

A number of conditions, sequence parameters, events, English text, states or similar terms are printed with the first letter of each word in uppercase and the rest lowercase (e.g., In, Out, Selective Reset, Bi-directional, Bus Control, Operation Response). Any lowercase uses of these words have the normal American-English meaning.

4 General description

4.1 Application environment

The master (controller) provides control of one to eight slaves (disk drives) with a device level set of commands and data transfers. The master and slave of a level 2 environment are the slave and facility of a level 3 environment. There are no facilities in the level 2 environment.

4.2 Logical Interface characteristics

4.2.1 Slave Control

Control of a slave is exerted by the master through the Bus Controls that initiate the Bus Exchanges for command, response, and data transfers. The Bus Controls incorporate coding that identifies the particular command, response, or data type. The Command Control codes specify the command type being sent to the slave, whereas the Response Control codes specify the response to be read from the slave. The Data Control codes specify the operation on fields of a disk sector and their orientation with respect to the start of a sector.

4.2.2 Information transfer

The interface uses the Double Octet mode to transfer 16-bit information. Data is transferred in the Data Streaming mode; Commands and Response are transferred in the Interlocked mode.

4.2.3 Data transfers

The master reads and writes on the disk by issuing Data Controls while staying oriented with the rotation of the disk. No addresses or octet counts are involved with the Data Controls. Operations are on whole fields and include field read, write, and skip operations. Field sizes include any master-managed Error Correcting Codes (ECC) and Cyclic Redundancy Codes (CRC).

4.2.4 Format Control

The organization of sectors and the number of sectors per track is controlled by a Format Specification, which may be mutually derived by both the slave and the master. This specification dictates the number of fields per sector, the length of each field, and the number of sectors per track.

4.3 Responsibility

4.3.1 Master

The master performs the following functions:

- Data buffering;
- Data management — Interleaving;
- Header management;
- Defect management — defect skipping, revectoring, spare sectors;
- Error detection and/or correction — Retries, ECC, CRC;
- Command generation.

4.3.2 Slave

The slave performs the following functions:

- Sectoring — Hard, soft, or both; variable or fixed;
- Formatting — Multiple fields, variable or fixed;
- Gap management — Phase Lock Oscillator (PLO), pads, interfield gaps, write-read recovery;
- Read Gate and Write Gate control;
- Phase Lock Oscillator (PLO), bit, and octet synchronization and pattern writing;
- Header verification for write (Format 1 only);
- Command execution;
- Response generation;
- Error detection and/or correction (optional);
- Dual port (optional).

5 Physical Interface Considerations

The Physical Interface is used as defined in ISO/IEC 9318-1, with the interpretations and additions defined in the following sub clauses.

5.1 Information transfer

5.1.1 Octet Mode

All Information Transfers (Commands, Responses, and Data) are transmitted in the double octet (16-bit) mode. Even-numbered octets are transmitted on BUS A and odd-numbered octets are transmitted on BUS B; i.e., the octet on BUS A is considered as being transmitted before the octet on BUS B.

For Command and Response information, bit 7 of an octet is the most significant bit (MSB) and bit 0 is the least significant bit (LSB). The most significant octets are transferred first in multioctet numeric parameters.

For data information, there is no definition of significance to the ordering of bits or octets. The order in which the octets are read is the same as that in which they are written.

Bit 7 on BUS A is the first recorded bit on the disk.

5.1.2 Transfer Mode

Command/Response information is always transferred in the Interlocked mode. Data information is always transferred in the Data Streaming mode. The transfer modes cannot be changed.

The Request Transfer Settings octet response reflects capabilities and settings of command/ response transfers in Double Octet and Interlocked modes.

5.1.3 Termination of transfers

The Information Transfer sequence may be terminated by either the slave or master.

5.1.4 Data streaming

Data streaming enables high transfer rates over long cables and is well suited to data transfers by synchronous disk. This is accomplished by not interlocking the SYNC IN and SYNC OUT, which eliminates a round-trip transmission delay.

Data streaming shall be used for data transfers as defined in ISO/IEC 9318-1, with the following interpretations:

- The SYNC IN pulses occur at the data rate of the disk and are generated at the rate of one per double octet.

- The nominal duty cycle of the SYNC IN is 50%.
- Upon recognizing a SYNC IN, the master returns a SYNC OUT after a delay not exceeding one double octet time at the maximum cable length measured at the slave connector.

During data output transfers, the slave shall prefetch data by initiating the SYNC IN pulses *n* octet times prior to the writing of the first data octet on the disk in order to account for transmission and master delays.

The value of *n* is dependent on master, slave, and transmission delays and the data rate of the slave. The slave shall require at least an *n* octet first in, first out (FIFO) buffer to account for delays less than maximum (*n* octet times). If data is not available in the slave when it must be written on the disk, a data underrun condition exists, the transfer is terminated and the Successful Information Transfer bit in the Slave Status octet may be set to zero.

Drives that support ECC/CRC and have that feature enabled shall complete the sector and write uncorrectable ECC/CRC if a data underrun condition occurs during a disk write.

5.2 Bus Octets

The bus octets defined in ISO/IEC 9318-1 shall be used as described, with the additions described in this sub clause. All valid state transitions shall be implemented by the slave.

5.2.1 Unsupported Bus Octets

The following bus octets are not supported:

- Facility Selection. Slave does not acknowledge selection;
- Request Facility Interrupts. Slave does not acknowledge request for facility interrupts;.
- Bus Acknowledge. Slave transmits zeroes on BUS B.

5.2.2 Bus Control Octet

The Bus Control octets define the direction and type of information transfer in bits 7-6 and the command to be executed in bits 5-0.

Bit	Description
7	0 = Operation Command/Operation Response 1 = Data
6	0 = Information Out 1 = Information In
5-0	Encoded Bus Control Command and Response Controls; see 7.1 Data Controls; see 7.2 and 7.3.

5.2.3 Master Status Octet

If the Master Status Octet indicates an unsuccessful information transfer following a Read Status Response, the slave shall not clear the pending Status Response conditions.

Bit	Description
7	Successful Information Transfer
6	Bus Parity Error
5-0	0

5.2.4 Slave Status Octet

The Slave Status provides information to the master on the results of the preceding Information Transfer in bits 7-5 and the command in bits 4-0 and 6. See 8.1 for further details on the Slave Status octet.

Bit	Description
7	Successful Information Transfer
6	Bus Parity Error
5	Odd Octet Transfer
4	Time-Dependent Operation
3-0	Encoded Operation Ending Status

5.2.5 Request Interrupts Octet

The organization of the Request Interrupt Octet is as follows:

Bit	Description
7	0
6	Report Busy Status: All slaves that are Busy shall place their bit- significant Address Octet Response on BUS B.
5	Report Ready Status: All slaves that are Ready shall place their bit-significant Address Octet Response on BUS B.
4	Master Power Fail Alert: The master is informing the slaves that it has detected that power is failing. The slaves shall acknowledge by placing their bit-significant Address Octet Response on BUS B after they have taken appropriate action to permit a graceful termination of activity.
3	Power On Status Request: All slaves with power on (but not necessarily ready) shall place their bit-significant Address Octet Response on BUS B.
2	Status Pending Interrupt: Slaves with Class 3 Interrupt pending shall respond by placing their bit-significant Address Octet Response on BUS B.
1	RPS Interrupt: Slaves with Class 2 Interrupt pending shall respond by placing their bit-significant Address Octet Response on BUS B.
0	Command Completion Interrupt: Slaves with Class 1 Interrupt pending shall respond by placing their bit-significant Address Octet Response on BUS B.

The slave shall respond to a Request Interrupt Octet with a bit-significant Address Octet Response (defined in 5.2.6).

5.2.5.1 Types of Interrupts

The slave asserts three interrupt classes defined in ISO/IEC 9318-1. The assertion of interrupt bit settings upon receipt of a Request Interrupts or Request Slave Interrupts sequence shall not be disabled by the Load Slave Functions.

5.2.5.1.1 Class 1 Command Completion Interrupt

This interrupt is asserted when a command is completed successfully and the Time-Dependent Operation bit of the Slave Status following the transfer of the command was set. The Command Completion Interrupt is not asserted for a Load Position command if the RPS Interrupt is to be asserted because of a valid RPS Target Sector Address parameter. The assertion of a Command Completion Interrupt is disabled by any of the following:

- Slave accepts any Bus Control;
- Slave is reset with a Selective Reset octet from this port with bit 1 (Reset Logical) set;
- Slave is reset with a Selective Reset from either port with bit 2 (Reset Slave) set.

5.2.5.1.2 Class 2 RPS Interrupt

This interrupt is asserted on each disk revolution during the time specified by the Load Position command or Load RPS Target Sector Address command. The assertion of the RPS Interrupt shall be disabled by any of the following:

- Setting X'FFFF' as the RPS Target Sector Address parameter;
- Slave accepts a Data Control;
- Slave accepts a Load Format Specification;
- Slave is reset while in neutral with a Selective Reset from either port with bit 1 (Reset Logical) set. If switched to the alternate port the assertion of the RPS Interrupt shall not be affected;
- Slave is reset with a Selective Reset from either port with bit 2 (Reset Slave) set.

5.2.5.1.3 Class 3 Status Pending Interrupt

This interrupt is asserted when there is a status exception in the Status Response that must be reported to the master. The assertion of Status Pending Interrupt is disabled by any of the following:

- Read Status Response with a Successful Information Transfer bit set in Master Status;
- Slave is reset with a Selective Reset Octet from this port with bit 1 (Reset Logical) set;
- Slave is reset with a Selective Reset Octet from either port with bit 2 (Reset Slave) set.

5.2.6 Address Octet Response

The slave shall respond to a Request Interrupt Octet or Selection Octet with an Address Octet Response that has the following organization:

Bit	Description
7	Slave 7
6	Slave 6
5	Slave 5
4	Slave 4
3	Slave 3
2	Slave 2
1	Slave 1
0	Slave 0

5.2.7 Selective Reset Octet

The receipt of this octet causes a Selective Reset of the type defined in bits 3-0. The following Reset Controls are defined:

Bit	Description	
7	Indicates Selective Reset Octet	
6-4	Slave address	
3	Slave Release	The slave shall release its interface drivers in the same manner as it would upon recognition of the MAINT state. The drivers are to remain released on the port over which the Reset was received until recognition of another Reset with this bit cleared.
2	Reset Slave	The slave shall be reset in the same way as when power on.
1	Reset Logical Interface	The Logical Interface for the port shall be reset. The slave's common resources that are not switched to the alternate port shall be reset. The Format Specification is not affected.
0	Reset Physical Interface	The Physical Interface for the port shall be reset.

5.2.8 Selection Octet

The Selection Octet bits 6-4 specify the slave address. Bits 3-1 are not supported and shall be set to zero. If bit 0, Priority Select, is set, the slave shall cease any operations with the alternate port, cancel any reserve on the alternate port, and attempt to honor selection from the selecting port.

Bit	Description
7	0
6-4	Slave Address
3-1	0
0	Priority Select

5.2.9 Request Transfer Settings Octet

The slave shall respond with a Transfer Settings Octet when it receives a Request Transfer Settings Octet that has the following organization:

Bit	Description
7	1
6-4	Slave Address
3-0	0

5.2.10 Slave Transfer Settings Octet

The slave shall respond to a Request Transfer Settings Octet from the master with a Transfer Settings Octet that has the following organization:

Bit	Description
7	0 = Transfer Settings Octet
6	0/1 = Maintenance Mode 1/2
5	1 = Double Octet Mode
4	0 = Interlocked Transfer
3	0 = Slave is not capable of Data Streaming transfers
2	1 = Slave is capable of Interlocked transfers
1	1 = Slave is capable of transferring in Double Octet Mode
0	0 = Slave is not capable of transferring in Single Octet Mode

The transfer settings reported are those reflecting the capabilities and settings for the slave during Command/Responses Information Transfers.

5.2.11 Request Slave Interrupts Octet

The slave shall respond with a Slave Interrupts Octet when it receives a Request Slave Interrupts Octet. The organization of a Request Slave Interrupts Octet shall be as follows:

Bit	Description
7	1
6-4	Slave address
3	1
2-0	0

5.2.12 Slave Interrupts Octet

The slave shall respond with a Slave Interrupts Octet when it receives a Request Slave Interrupts Octet. The organization of the Slave Interrupts Octet shall be as follows:

Bit	Description
7	Reserved, set to zero.
6	Busy: Busy is an indication that the slave is unable to respond by the master's request; i.e., it is currently executing a Bus Control (previously issued from this port or the alternate port), or is reserved by the alternate port.
5	Ready: Ready is an indication that the slave is operational and able to accept any supported Command or Data Bus Control code that is in context.
4	Reserved, set to zero.
3	Priority Selected: This bit shall be set when a successful Priority Select has occurred during selection of the alternate port. This bit shall be reset at the end of the Request Slave Interrupts sequence during which it was reported.
2	Status Pending Interrupt: This interrupt shall be asserted when there is a status exception to be reported in the Status Response.
1	RPS Interrupt: This interrupt shall be asserted during the time on each disk revolution that the RPS Pulse is valid (the data heads are in the requested area defined by the RPS Target Sector Address and optional parameters, if any).
0	Command Completion Interrupt: This interrupt shall be asserted when a command, previously transferred with the Time-Dependent Operation bit set in the Slave Status octet is completed successfully. The Command Completion Interrupt is not asserted on a Load Position command that includes a valid RPS Target Sector Address parameter.

5.3 ATTENTION IN signal

The ATTENTION IN signal shall be asserted by an unselected slave when the RPS, Command Completion or Status Pending Interrupts are asserted, or if the slave is enabled to assert ATTENTION IN upon "No Longer Busy" (a Busy to Not Busy condition occurs). See 5.2.5.1 for a description of interrupts.

The Load Slave Function command provides for enabling and disabling the individual interrupts from asserting ATTENTION IN, on a port basis.

All ATTENTION IN assertions shall be enabled upon power-on Reset or Selective Reset (Reset Slave) except for "No Longer Busy".

6 Slave Functions

6.1 Disk Format

There are three possible disk formats: Format 1 Sector Mode 1, Format 1 Sector Mode 2, and Format 2. At least one format shall be supported.

6.1.1 Format 1 – Fixed Block Format

The fixed block format allows for a slave to be formatted with a variable number of identical sectors per track. The length and organization of a fixed block sector shall be determined by the Fixed Block Format Specification. The fixed block format allows the slave to operate in one of the following two sector modes:

- Sector Mode 1. Every sector is of the same length and organization, which is fixed by the slave;
- Sector Mode 2. Every sector is of the same length and organization, but the length and organization is programmable.

The start of a fixed block sector may be determined by one of the methods described in 6.1.1.1 and 6.1.1.2.

6.1.1.1 Fixed Block Hard Sector

A slave in the hard-sectored mode typically determines the start of each sector by counting the number of octets from the index mark at the beginning of a track. The number of sectors is specified by the Fixed Block Format Specification.

6.1.1.2 Fixed Block Soft Sector

A slave in the soft-sectored mode uses Address Marks to mark the start of each sector. The number of sectors is specified by the Fixed Block Format Specification. Each track of the slave shall be formatted with Address Marks by the Perform Sector Marking function.

6.1.2 Format 2 – Variable Block Format

The variable block format allows for a slave to be formatted with a variable number of variable length sectors per track. Each sector may have a different organization.

A slave operating with a variable block format uses Address Marks to mark the start of each sector. The variable block format depends on the controller to determine the length and number of data fields in a sector during a disk write operation.

6.1.3 Format Control

The slave shall be responsible for format control. It shall be capable of performing the following functions in meeting the requirements of the appropriate Format Specification:

- Determining the start of each sector on the track;
- Determining the start of each field within a sector;
- Counting the number of octets in each field of a sector;
- Activating Read Gate and Write Gate in accordance with the current Data Control and the Format Specification;
- Acquiring PLO, bit, and octet synchronization on read operations;
- Writing of PLO, bit, and octet synchronization patterns.

6.2 Format Specification

The Format Specification is an ordered list of parameters that define for the slave how the tracks and sectors of the disk are to be organized. It is transferred to the slave by the Load Format Specification control, and the master may retrieve it by the Read Format Specification control. Both the master and the slave may establish the parameters during the initialization of the Format Specification.

6.2.1 Format initialization

The Format Specification is not initialized when the slave is powered on, unless the slave has a means of saving it. The master may save the current Format Specifications and load them into the slave. (See annex C for description of storage of Format Specifications).

A Format Specification with its Initialized bit reset is not complete and can not be used by the slave for format control or data transfer.

A Format Specification shall be initialized by the following procedure:

a) The master loads a Format Specification with its Initialized bit reset to invalidate the old specification, if any, and to condition the slave to start the initialization process. The master provides any of the parameters that it is to contribute to the Format Specification. Parameters not specified by the master shall be X'FFFF'.

   Alternatively, the master loads a completely specified Format Specification with the Initialized bit reset.

   If the slave determines that there are missing or incorrect parameters, it sets the appropriate Command Exception status bit.

b) The master reads an initialized Format Specification from the slave to obtain the slave's fixed parameters, calculated parameters, or both. The master shall check the Format Specification against its requirements and adjust its own control to the specification.

6.2.2 Manufacturer's Format Specification

The slave shall have a built-in Manufacturer's Format Specification, which is used primarily to write and read the Defect Map. It shall be invoked by the Load Format Specification control with Bit 6 of the Flag Octet set, and shall remain in effect until another Load Format Specification control is accepted by the slave. It may be read by a Read Format Specification control when it is invoked.

The format shall provide for a header and one or two data fields, each field not exceeding 1024 octets in length. The header contents, the CRC, and the ECC are not specified. The format shall be organized to allow for either sector type or field type of controls. There shall be no sector interleaving.

The master shall be capable of recovering the data from data fields formatted with this specification, without the use of the headers. Sectors shall be read in their physical order from the Index.

The Manufacturer's Format Specification may be a Fixed Block or Variable Block Format Specification as indicated by the Format Type Code.

Slaves that operate only in the Variable Block Format shall be capable of reporting the Manufacturer's Default Format in a Fixed Block format so that the master may determine the number of sectors per track, fields per sector, and octets per field.

6.2.3 Fixed Block mode

A mode of operation whereby certain slaves, using the Fixed Block Read/Write bus control codes set and Format Specification, allow the master to operate on disk data sectors of a fixed organization.

This mode of operation is mutually exclusive with the Variable Block Mode described in 6.2.4.

6.2.3.1 Fixed Block Format Specification

The Fixed Block Format Specification allows for sectors of one or more fields, but all sectors shall have the same organization. The Fixed Block Format Specification has the following format:

Octet	Bit	Description
0-1		Number of octets following: equals n - 1
2		Format type code: 1 = fixed block
3		Flag octet
	7	Initialized (Format Specification Initialization complete)
	6	Manufacturer's Default Format Specification
	5	Sector mode 2
	4	Sector mode 1
	3	Use soft sectoring
	2	Use hard sectoring
	1	Use field Data Controls
	0	Use sector Data Controls
4-5		Number of sectors per track
6-9		Number of physical octets per sector
A-B		Number of beginning header octets to be skipped during header verify
C-D		Number of fields per sector (m+1)
E-11		Number of octets per field 0 m=0
12-13		Master turnaround delay 0 m=0
(n-5):(n-2)		Number of octets per field m
(n-1):n		Master turnaround delay m

NOTE — m equals the field number of the last field in a sector. Fields are numbered starting from zero. n is the octet number of the last octet transferred.

6.2.3.1.1 Number of octets

This double-octet parameter contains the octet count for the Format Specification. It does not include itself in the octet count.

6.2.3.1.2 Format Type Code

This single-octet parameter contains the format type code. If this parameter equals 1, then the Format Specification is a fixed block type and is organized as such.

6.2.3.1.3 Flag Octet

This single-octet parameter contains the flag information for the Fixed Block Format Specification and indicates if the Format Specification is initialized, if the Format Specification is a manufacturer's default format or user's format, which sector mode is to be used, and what type of data controls the master intends to use with this format.

6.2.3.1.4 Number of sectors per track

The number of sectors per track shall be determined by one of the following schemes:

- Slave with a Fixed Sector: If the slave has a fixed length sector size, the slave enters this number in the Format Specification every time it is read by the master;

- Defined by the Master: If the master is to define the number of sectors per track, the master sets the number in the Format Specification every time it is transmitted to the slave. The master specifies as X'FFFF' at least one of the remaining three quantities: the number of fields per sector, one or more field sizes, or one or more of the Master Turnaround Delays. The slave computes the value of the unspecified quantity;

- Calculated by the Slave: If the slave is to calculate the number of sectors that can be fitted on the track, the master may then supply it with the number of fields, the field sizes, and the Master Turnaround Delays. The master specifies the number of sectors as X'FFFF'. The slave then enters the number of sectors in the Format Specification.

In addition to the values that may be specified, the master has to take into consideration the number of slave-controlled overhead octets in the gaps.

6.2.3.1.5 Number of physical octets per sector

The number of physical octets per sector may be supplied by the master or set to X'F..F' to be computed by the slave. The number of physical octets per sector is typically computed by the slave and is used by the master to determine the sector number of a defect specified by the Defect Map. It may also be used to infer the size of the field gaps.

6.2.3.1.6 Number of octets to be skipped during header verify

This parameter shall be supplied by the master to specify the number of octets, starting from the beginning of the header, to be skipped in the transfer of data and the slave's verification of the header during the execution of a Verify Header Data Control.

6.2.3.1.7 Number of fields per sector

The number of fields per sector may be supplied by the master or set to X'F...F' and be computed by the slave. The first field in a sector is usually the header identification.

6.2.3.1.8 Number of octets per field

The number of octets in each field may be supplied by the master based on system considerations or be set to X'F...F' and be computed by the slave. Master-managed ECC and CRC are considered as part of the field length for the user. Zero is not a valid value for a specified field.

6.2.3.1.9 Master Turnaround Delay

The Master Turnaround Delay can be supplied by the master or set to X'F...F' to be computed by the slave.

The Master Turnaround Delay is the time, measured in octets, required by the master between fields to handle the Ending Status sequence from one Data Control and to initiate another. It is measured at the slave, and therefore includes transmission delays, from the point in time when the slave terminates a transfer (drops SLAVE IN) until the next Data Control is received at the slave. The actual gap used by the slave is the Master Turnaround Delay, increased to account for its own delays including: read propagation delay, transfer termination time, Slave Status turnaround time, and Data Control decoding.

The master shall provide this parameter for each of the fields. If no Data Control is to be sent to the slave after a field, the parameter for that field may be zero. This parameter and the Field Size parameter are repeated until every sector field is defined.

6.2.4 Variable Block Mode

A mode of operation whereby certain slaves, using an alternate Read/Write bus control code set and Format Specification command, allow the master to write sectors with any number of data fields of any length, up to the track capacity of the slave. This mode of operation is mutually exclusive with the Fixed Block Mode described in 6.2.1.

6.2.4.1 Variable Block Format Specification

Variable Block Mode uses the following Format Specification to define the track format.

Octet	Bit	Description
0-1		Number of octets following: equals X'16'
2		Format type code: 2 = variable block
3		Flag octet
	7	Initialized (Format Specification Initialization complete)
	6	Manufacturer's Default Format Specification
	5-0	Reserved, set to zero
4-5		Cell length in octets
6-7		RPS divisions per track
8-9		Index Mark Gap length in cells
A-B		Home Field Data Segment length in cells
C-D		Home Field Gap length in cells
E		Home Field Skip length in cells
F		Home Field retries
10-11		Header Field Data Segment length in cells
12-13		Header Field Gap length in cells
14-15		Data Field Gap length in cells
16-17		Defect Skip length in cells

6.2.4.1.1 Number of octets

This double-octet parameter contains the octet count for the Format Specification. It does not include itself in the octet count.

6.2.4.1.2 Format Type code

This single-octet parameter contains the format type code. If this parameter equals 2, then the Format Specification is a variable block type and is organized as such.

6.2.4.1.3 Flag Octet

This single-octet parameter contains the flag information for the Variable Block Format Specification and indicates if the Format Specification is initialized, and if the Format Specification is a manufacturer's default format or a user's format.

6.2.4.1.4 Cell length

This double-octet parameter defines the length of a Cell in octets. Most formatting elements are defined in terms of Cells. The Cell length shall be longer in time than a Master Termination Delay (twice the Cable Delay plus the Master Turn-around Delay).

6.2.4.1.5 RPS Divisions per track

This double-octet parameter defines the number of RPS Divisions per track.

In the Variable Block Mode, RPS Divisions have no relationship with the location of the Data Fields and are only used to cause the assertion of RPS Interrupts and to report the current head position for certain response bus controls.

6.2.4.1.6 Index Mark Gap length

This double-octet parameter defines the length in cells of the gap following the Index Mark. This gap lies between the Index Mark and the beginning of the Home Field.

6.2.4.1.7 Home Field Data Segment length

This double-octet parameter defines the length in cells of the Home Field Data Segment. This field may be set to zero if the master wishes to have no Home Field.

6.2.4.1.8 Home Field Gap length

This double-octet parameter defines the length in cells of the Home Field Gap. This gap shall be large enough to allow for all re-command delays (e.g., master decision delay, cable delays) between the Home Field Data Segment and any Header or Data Field that follows. This gap shall also account for head-position uncertainty due to head scatter and system cable configuration signal delay difference. This field may be set to zero only if the Home Field Data Segment is set to zero.

6.2.4.1.9 Home Field Skip Length

This single-octet parameter defines the length in cells to be skipped, when a Home Field read fails, before another Home Field read is to be attempted by the slave. This field may be set to zero.

6.2.4.1.10 Home Field retries

This single-octet parameter defines the number of automatic Home Field read retries that the slave is expected to perform when performing a Read Home Field operation. This field may be set to zero.

6.2.4.1.11 Header Field Data Segment length

This double-octet parameter defines the length in cells of the Header Field Data Segment.

6.2.4.1.12 Header Field Gap length

This double-octet parameter defines the length in cells of the Header Field Gap. This gap shall be large enough to allow for all re-command delays (e.g., master decision delay, cable delays) between the Header Field Data Segment and any Header or Data Field that follows. This gap shall also account for head-position uncertainty due to head scatter and system cable configuration signal delay difference.

6.2.4.1.13 Data Field Gap length

This double-octet parameter defines the length in cells of the Data Field Gap segment. This gap shall be large enough to allow for all re-command delays (e.g., master decision delay, cable delays) between the Data Field Data Segment and any Header or Data Field that follows. This gap shall also account for head-position uncertainty due to head scatter and system cable configuration signal delay difference.

6.2.4.1.14 Defect Skip Segment length

This double-octet parameter defines the length in cells of the Defect Skip Segment. This segment is used in place of the larger Data Field Gap segment when ending certain write operations to allow for skipping over small defects. This segment shall be large enough to allow for all re-command delays (e.g., master turnaround delay, cable delays) between the Data Field Data Segment and the following Data Field. This segment shall also account for head-position uncertainty due to head scatter and system cable configuration signal delay difference.

6.3 Slave conditions

The manner in which a slave responds to a command shall be determined by its condition.

The condition of a slave is affected by its intrinsic as well as its operational characteristics. The conditions are:

- P-Available: The port is capable of responding on the Physical Interface;
- Not P-Available: The port does not respond on the Physical Interface;
- Operational: The slave is capable of responding on the Physical Interface and executing commands;
- Ready: Ready is an indication that the slave is operational and able to accept any supported command or data control code that is in context;
- Not Ready: The slave is not able to execute any supported command or data control code in context that requires access to the drive Head Disk Assembly (HDA);
- Status Pending: The slave has a Status Response for the master. The Status Pending Interrupt is asserted;
- Active: The slave has accepted a command and has not yet generated the corresponding Slave Status with the Time-Dependent Operation (TDO) bit reset nor asserted the Command Completion Interrupt;
- Inactive: The slave has no command outstanding, but is capable of receiving one from the master;
- Reset Slave: The slave is in an initialized condition when it has no cognizance of past events. This condition can come about as a result of an external reset by the master, an internal initialization (e.g., power-on), or an unsuccessful internal recovery attempt from a severe slave error.

Slaves provide predictable status on their ability to process commands. This information is obtained via the Request Interrupts sequence (see table 1).

An Operational slave is able to respond to Master Reset, Selective Reset, Request Interrupts, Request Slave Interrupts, Request Transfer Settings and any Bus Control Codes that are in context.

A Busy slave may or may not be able to handle Selection, Deselection, Bus Control, and Ending Status sequences depending on how the optional dual port is implemented. For example:

- A physical level dual port switch shall indicate Busy during Selection;
- A logical level dual port switch shall honor the Selection, Deselection, and Bus Control sequences, and indicate Busy in the Slave Status Octet of the Ending Status sequence.

Table 1 — Slave Conditions

Power	Ready	Busy	Description
0	0	0	Nothing happening at the slave (Not P-Available)
0	1	1	Illegal Condition
0	1	X	Illegal Condition
1	0	0	The slave is Operational but Not Ready. It is able to handle Selection, Deselection, Bus Control and Ending Status Sequences. All Read/Write Data Controls and any Command/Response Bus Controls that require the drive Head Disk Assembly (HDA) to be accessed shall be rejected with a Slave Status of Operation Exception, Bus Control Out of Context.
1	1	0	The slave is operational and Ready. It is able to handle Selection, Deselection, Bus Control and Ending status sequences, and Information Transfers in context.
1	0	1	The Slave is Operational, Not Ready, and Busy.
1	1	1	The Slave is Operational, Ready, and Busy.

6.4 Dual Port (optional)

The optional dual port consists of manual and programmed Enable/Disable controls for a port (static switching) and the logical constructs for dynamically switching of the slave between two ports. The switching of a slave to a port creates an allegiance of the slave's common resources (slave dependent) to that port.

6.4.1 Port Enable/Disable

Ports may be individually enabled or disabled by command and by optional manual controls. When a port is disabled, it is made Not P-Available on the Physical Interface. The disabling of a port may be either orderly or destructive with the choice being implementation dependent. Both ports are enabled on power-on, if not disabled by manual means.

The orderly disabling of a port takes effect when the Physical Interface on that port is in the IDLE state and the slave, if switched to the port, is inactive. The destructive disabling of a port takes place immediately without regard to the state of the interface.

The enabling of a port takes effect when the Physical Interface on that port is in the IDLE state.

If a port is disabled by command, changing a manual port switch from Disable to Enable shall cause the port to be Enabled. However, the disabling of a port by a manual switch cannot be overridden by a command.

The disabling of a port, either by command or switch, shall cause any explicit or implicit reservation to be cleared and shall cause any solicited status associated with the disabled port to be reset.

6.4.2 Slave Accessibility mode

The slave can appear in one of two accessibility modes: Neutral and Switched.

6.4.2.1 Switching

Following a power-on Reset or Selective Reset (Reset Slave) condition, the slave enters the Neutral mode. While in Neutral mode, the slave can be accessed via either enabled port and can become switched to an enabled port, either implicitly or explicitly.

-Implicit Switching: A slave becomes switched to a port implicitly under the following two conditions:

1)Communication (Selection or Bus Exchange, as determined by the slave) over a port is initiated in accordance with the Physical Interface protocol described in ISO/IEC 9318-1. The slave returns to the Neutral mode when the communication ends, unless returned to Neutral by a Reset.
2)The slave accepts a command. The slave returns to Neutral when the command is completed, unless returned to Neutral by a Reset.

- Explicit Switching: Explicit allegiance occurs when a Reserve or Priority Reserve function is issued to the port. The slave becomes switched to the port over which the Reserve is received. It remains switched until a Release function is received, the alternate port receives a Priority Reserve or Priority Select, or the slave is reset with a Selective Reset Octet received at either port with bit 2 (Reset Slave) set.

6.4.2.2 Switch mechanism

The reserve mechanism is slave dependent. If a port is reserved, the alternate port is made Busy at either the Physical Interface or the Logical Interface.

If a slave rejects selection because it is switched to another port, then an ATTENTION IN shall be asserted when the slave becomes "No Longer Busy", if enabled to do so by the function code 1E (Enable "No Longer Busy").

The switch mechanism may be implemented within a slave in the following ways:

- A Physical Interface Only switch which does not permit a reservation during selection at the alternate port;
- A Logical Interface Only switch which can be implemented only if a slave provides the ability to process commands on both ports concurrently;
- A combination of both is also possible. In such a case, a Priority Select at the Physical Interface shall override any logical reservation.

6.4.3 Notification of alternate port exception condition

The occurrence of the following conditions on the alternate port shall be reported as an Unsolicited Exception in the Status Response:

- Priority Select;
- Format Specification changed;
- Reset complete;
- Receipt of the Notify Alternate Port of Format Complete Function Code (see 7.1.1.8).

6.4.4 Attention

When the slave has an allegiance to a port, ATTENTION IN may be asserted to the port to which the slave has the allegiance, provided that the interrupt condition is for that port, but not selected by that port.

See also 5.3.

6.5 Reset

6.5.1 External Reset

An external reset, in the form of a Master Reset or a Selective Reset received over the Physical Interface, can be presented by the master at any time regardless of the condition of the slave. It shall affect the port over which it is received and the slave, if not switched to the alternate port. (See also Selective Reset (5.2.7).

The Master Reset and the Reset Physical Interface type of Selective Reset resets the Physical Interface of the port over which they were received and do not affect logical conditions. (e.g. Reservation switch, pending interrupts, Format Specification, or RPS optional parameters).

The Reset Logical Interface type of Selective Reset resets the Logical Interface of the port over which it was received, including any pending interrupts. If switched to the other port the assertion of the RPS Interrupt is not affected. The Reset Logical does not affect the state of any port switch, the Format Specification, or the RPS optional parameters.

The Reset Slave type of Selective Reset resets the entire slave to its power-on condition. The Reset Slave does not affect the power state of the slave motor. Following the Reset, the slave enters the Operational condition, except when precluded from doing so by conditions that force it into a Not Operational condition.

6.5.2 Internal Reset

When a slave internal reset occurs, an Unsolicited Exception Status bit is set and the Status Pending Interrupt is activated and sent to all ports that are enabled.

6.6 Head Control

The slave's head addressing register is loaded by the Load Head Address and Load Position controls.

The head address register shall be capable of being incremented at the end of a successful data transfer, if bit 4 of a Fixed Block Data Control is set. A successful transfer requires that the Successful Information Transfer bit be set in both the Master Status and the Slave Status. The head address shall advance immediately upon receiving the Master Status Octet.

The address of the first head shall be zero and the address of the last head shall be one less than the number of heads specified in the Read Configuration response. The counter increments to zero after the last head address.

6.7 Rotational Position Sensing (RPS) (optional)

The RPS option provides the ability for the slave to assert (Class 2) RPS Interrupts.

6.7.1 RPS Target Sector Address

See 7.1.6.1.

6.7.2 RPS Interrupt

See 5.2.5.1.2.

6.8 Slave ECC (optional)

The slave may optionally perform its own ECC on any or all fields of the sector. A Ending Slave Status code is provided for notifying the master of a slave-detected data error. The Read Correction Vectors control is provided to allow the master to obtain the correction vectors from the slave. The ECC function can be disabled and enabled by Load Slave Function codes.

6.9 Power Sequencing (optional)

Power Sequencing provides a means for the master to sequentially start the motors of the slaves, so as to reduce the power line surge current while the slaves are reaching operating speed. The master shall issue a Spin Up function to each slave as it becomes operational. The Command Completion interrupt for this function shall not occur until the slave is up to speed.

This Slave Function is optional for the slave, or in some cases may be bypassed by a switch, if power sequencing is not required or provided by some other means. If power sequencing is supported by the slave and is enabled, the slave shall not turn on its motor when power is applied.

7 Bus Controls

Bus Controls specify the condition of the bus and the information to be transferred. The three types of Bus Controls are: Command, Response, and Data. The Bus Control Code Octet has the following general form:

Bit	Description
7	1 = Data Control 0 = Command/Response Control
6	1 = Information In (Read) 0 = Information Out (Write)
5-0	Operation Specific

7.1 Command/Response Controls

The Command/Response Controls are forms of the Bus Control Octet that allow commands to be transmitted to the slave and responses to be read from the slave. The command and response types are identified in the Bus Control coding. The command and response information transferred consists of ordered lists of parameters. Parameters are transmitted with the most significant octet first.

Commands shall not be stacked in the slave.

The valid Command/Response Controls and their hexadecimal codes are in the following table and are described in 7.1.1 through 7.1.15. All other Command/Response Control codes (including 00 and 40 used in IPI Level 3) are rejected as being invalid.

Bus Control Code	Command/Response Control
01	Load Slave Function
02	Load Format Specification
03	Load Slave Specific Information
04	Load Cylinder Address
05	Load Head Address
06	Load RPS Target Sector Address
07	Load Position
41	Read Configuration
42	Read Format Specification
43	Read Slave Specific Information
44	Read Status
45	Read Correction Vectors
46	Read Current Sector Address
47	Read Current Position
48	Read Extended Status

7.1.1 Load Slave Function (01)

The Load Slave Function bus control code causes the slave to perform the action specified by the various function codes defined by this command. The function code is carried in both octets of the double-octet parameter transmitted to the slave in the Information Transfer Sequence. If the two octets are not the same or the octets contain an invalid value, the command shall be rejected as containing an invalid parameter. The format of the command shall be as follows:

Octet Parameters	Description
0	Function code
1	Function code repeated

The various functions and their hexadecimal codes shall be as described in 7.1.1.1 through 7.1.1.45.

7.1.1.1 Reserved for future use (00-0F)

7.1.1.2 Disable Alternate Port (10) (optional)

This function code causes the slave to disable the alternate port, making it Unavailable at the Physical Interface. This function may cause an immediate and hard disabling of the alternate port. The function causes an exception if there is only one port.

7.1.1.3 Enable Alternate Port (11) (optional)

This function code causes the slave to enable the alternate port, thereby making it available at the Physical Interface. The function causes an exception if there is only one port.

7.1.1.4 Disable Port (12) (optional)

This function code causes the slave to disable the port over which it received this function, making it unavailable. This function takes effect when the master deselects from this port.

7.1.1.5 Priority Reserve (13) (optional)

This function code causes the slave to be reserved to this port, even if already reserved by the other port. The reserve remains in effect until released, until a Priority Reserve is executed by the alternate port, until a selection octet with the Priority Select bit set is issued by the alternate port, or until the slave is reset with a Selective Reset Octet from either port with bit 2 (Reset Slave) set.

7.1.1.6 Reserve (14) (optional)

This function code causes the slave to be reserved to this port. The reserve remains in effect until released, until a Priority Reserve is executed by the alternate port, until a selection octet with the Priority Select bit set is issued by the alternate port, or until the slave is reset with a Selective Reset Octet from either port with bit 2 (Reset Slave) set.

7.1.1.7 Release (15) (optional)

This function code causes the slave to be released from this port and to enter the Neutral Mode.

7.1.1.8 Notify alternate port of format completion (16) (optional)

This function code sets the Format Completed bit in the Status Response for the alternate port immediately. The function causes an exception if there is only one port.

7.1.1.9 Reserved for future use (17)

7.1.1.10 Disable successful Command Completion Interrupt (Class 1) attention (18)

This function code causes the slave to disable the Command Completion Interrupt from asserting ATTENTION IN at this port.

7.1.1.11 Enable successful Command Completion Interrupt (Class 1) attention (19)

This function code causes the slave to enable the Command Completion Interrupt to assert ATTENTION IN at this port. It is enabled after a power-on Reset or a Selective Reset with bit 2 (Reset Slave) of the Selective Reset Octet set.

7.1.1.12 Disable RPS Interrupt (Class 2) attention (1A)

This function code causes the slave to disable the RPS Interrupt from asserting ATTENTION IN at this port.

7.1.1.13 Enable RPS Interrupt (Class 2) attention (1B)

This function code causes the slave to enable the RPS Interrupt to assert ATTENTION IN at this port. It is enabled after a power-on Reset or a Selective Reset with bit 2 (Reset Slave) of the Selective Reset Octet set.

7.1.1.14 Disable Status Pending Interrupt (Class 3) attention (1C)

This function code causes the slave to disable the Status Pending Interrupt from asserting ATTENTION IN at this port.

7.1.1.15 Enable Status Pending Interrupt (Class 3) attention (1D)

This function code causes the slave to enable the Status Pending Interrupt to assert ATTENTION IN at this port. It is enabled after a power-on Reset or a Selective Reset with bit 2 (Reset Slave) of the Selective Reset Octet set.

7.1.1.16 Disable "No Longer Busy" attention (1E) (optional)

This function code causes the slave to disable the slave's "No Longer Busy" from asserting ATTENTION IN at this port. It is disabled after a power-on Reset or a Selective Reset with bit 2 (Reset Slave) of the Selective Reset octet set.

7.1.1.17 Enable "No Longer Busy" attention (1F) (optional)

This function code causes the slave to enable the slave's "No Longer Busy" to assert ATTENTION IN at this port.

7.1.1.18 No operation (20)

This function code performs no operation.

7.1.1.19 Reserved for future use (21)

7.1.1.20 Spin Up (22) (optional)

This function code causes the slave to turn on its motor. If it cannot perform this operation, an exception status is set in the Status Response. The Command Completion Interrupt is asserted when the slave is up to speed (or nearly so). If the slave motor is on, the Time-Dependent Operation bit of the Slave Status need not be used. Resets do not affect the power state of the slave motor.

7.1.1.21 Spin Down (23) (optional)

This function code causes the slave to turn off its motor. If it cannot perform this operation, an exception status is set in the Status Response. Resets do not affect the power state of the slave motor.

7.1.1.22 Load Heads (24) (optional)

This function code causes the slave to load its heads. If it cannot perform this operation, an exception status is set in the Status Response. Resets do not affect the load state of the slave heads.

7.1.1.23 Unload Heads (25) (optional)

This function code causes the slave to unload its heads. If it cannot perform this operation, an exception status is set in the Status Response. Resets do not affect the load state of the slave heads.

7.1.1.24 Lock Carriage (26) (optional)

This function code causes the slave to lock the carriage. If it cannot perform this operation, an exception status is set in the Status Response. Resets do not affect the lock state of the slave heads.

7.1.1.25 Unlock Carriage (27) (optional)

This function code causes the slave to unlock the carriage. If it cannot perform this operation, an exception status is set in the Status Response. Resets do not affect the lock state of the slave heads.

7.1.1.26 Recalibrate (28)

This function code shall cause the slave to move the heads to cylinder 0.

7.1.1.27 Execute Internal Diagnostic (29)

This function code causes the slave to execute its built-in diagnostics.

7.1.1.28 Reserved for future use (2A)

7.1.1.29 Perform Sector Marking (2B) (optional)

This function code causes the slave to format the currently selected track with Address Marks in accordance with the Format Specification. If the slave cannot perform this function or if the current Format Specification does not indicate that soft sectoring is to be used, an exception status is set in the Status Response.

7.1.1.30 Disable Slave ECC (2C) (optional)

This function code causes the slave to disable the slave ECC mechanism. If the slave cannot perform this operation, an exception status is set in the Status Response.

7.1.1.31 Enable Slave ECC (2D) (optional)

This function code causes the slave to enable the slave ECC mechanism. If the slave cannot perform this operation, an exception status is set in the Status Response.

7.1.1.32 Reserved for future use (2E-40)

7.1.1.33 Reset Offset (41)

This function code clears any head offset. If a slave does not support track offsets, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.34 Set Positive Offset 1 (42)

This function code causes the slave to offset its heads by a minimum value (slave dependent) in the positive direction (away from the spindle). If a slave does not support track offsets, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.35 Set Negative Offset 1 (43)

This function code causes the slave to offset its heads by a minimum value (slave dependent) in the negative direction (towards the spindle). If a slave does not support track offsets, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.36 Set Positive Offset 2 (44)

This function code causes the slave to offset its heads by a value greater than Positive Offset 1 (slave dependent) in the positive direction. If a slave does not support track offsets, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.37 Set Negative Offset 2 (45)

This function code causes the slave to offset its heads by a value greater than Negative Offset 1 (slave dependent) in the negative direction. If a slave does not support track offsets, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.38 Set Positive Offset 3 (46)

This function code causes the slave to offset its heads by a value greater than Positive Offset 2 (slave dependent) in the positive direction. If a slave does not support track offsets, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.39 Set Negative Offset 3 (47)

This function code causes the slave to offset its heads by a value greater than Negative Offset 2 (slave dependent) in the negative direction. If a slave does not support track offsets, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.40 Set Normal Strobe (48)

This function code causes the slave to set normal data strobe. If a slave does not support early/late data recovery strobes, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.41 Set Early Strobe (49)

This function code causes the slave to set an early data strobe. If a slave does not support early/late data recovery strobes, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.42 Set Late Strobe (4A)

This function code causes the slave to set a late data strobe. If a slave does not support early/late data recovery strobes, it shall respond with an immediate Command Complete Ending Status Octet upon receipt of this function code and set bits at Extended Status Response as if executed.

7.1.1.43 Reserved for future use (4B-7F)

7.1.1.44 Perform Slave Specific Function n (8n) (optional)

This function code causes slave-specific function n to be performed. If it can not be performed, the function is rejected or causes an exception.

7.1.1.45 Reserved for future use (90-FF)

7.1.2 Load Format Specification (02)

The Load Format Specification command control transmits a Format Specification to the slave. The Format Specification is described in 6.2. The format of the command parameters shall be as follows:

Octet	Parameters
0-1	Number of octets following: equals n - 1
2-3	Format Type and Flag Octet
4-n	Remainder of Format Specification (optional)

If bit 6 of the Flag Octet is set, the Manufacturer's Format Specification is invoked and no more parameters need be supplied by the master.

7.1.3 Load Slave-Specific Information (03)

The Load Slave-Specific Information command control transmits slave-specific information (e.g., diagnostics) to the slave. The format of the command parameters shall be as follows:

Octet	Parameters
0-1	Number of octets following: equals n - 1
2-n	Slave-specific information

7.1.4 Load Cylinder Address (04)

The Load Cylinder Address command control causes the slave to seek to the cylinder specified in the four octets transmitted to the slave. RPS Interrupts, if enabled, are not asserted until the slave is on cylinder. Any head or strobe offset is cleared. The format of the command parameters shall be as follows:

Octet	Parameter
0-3	Cylinder address

7.1.5 Load Head Address (05)

The Load Head Address command control causes the slave to select the head specified in the two octets transmitted to the slave. Any head or strobe offset is cleared. The format of the command parameters shall be as follows:

Octet	Parameter
0-1	Head address

7.1.6 Load RPS Target Sector Address (06)

The Load RPS Target Sector Address command control causes the slave to select the physical sector (Fixed Block Mode) or RPS division (Variable Block Mode) specified in the two octets transmitted to the slave for the RPS Target Sector Address parameter.

The RPS Interrupt is disabled if the RPS Target Sector Address parameter is equal to X'FFFF'.

There are several optional parameters which may be supported by the slave with this bus control code. The slave shall terminate after the last parameter it supports. The master shall be prepared to master-terminate after the last parameter it transfers.

If an optional parameter is not supplied by the master, the value shall remain at its previous setting.

The format of the command parameters shall be as follows:

Octet	Parameter
0-1	RPS Target Sector Address
2-3	RPS Pulse Width Extension(Optional)
4-7	RPS Pulse Width(Optional)
8-B	RPS Pulse Width Skew(Optional)

7.1.6.1 RPS Target Sector Address

This double-octet parameter is the address of the physical sector (Fixed Block Mode) or RPS division (Variable Block Mode). This physical location shall equal the RPS Pulse unless modified with the optional parameters.

If the RPS option is supported, the RPS Target Sector Address is set to a value of X'FFFF' at power-on Reset or Selective Reset (Reset Slave) and shall be reset to X'FFFF' if a Format Specification is accepted by the slave.

If the RPS Target Sector Address option is not supported, the RPS Target Sector Address is equal to X'0000'.

7.1.6.2 RPS Pulse Width Extension

This double-octet parameter may be transferred in addition to the previous octets to extend the RPS Pulse. For the Fixed Block Mode, this parameter specifies the number of sectors that the RPS Pulse is extended. For the Variable Block Mode this parameter specifies the number of RPS Divisions that the RPS Pulse is extended. The RPS Pulse is extended by this parameter in the direction ahead of the target. If this parameter contains a value of X'0000', then the RPS Pulse defaults to a length of one sector or RPS division.

If the following RPS Pulse Width parameter is used this parameter acts as a multiplier to produce the total RPS pulse width.

The RPS Pulse Width Extension parameter is set to a value of X'0000' at power-on Reset or Selective Reset (Reset Slave) and shall be reset to X'0000' if a Format Specification is accepted by the slave.

7.1.6.3 RPS Pulse Width (Format 1 option)

This four-octet parameter may be transferred in addition to the previous octets to define the RPS pulse width in octets. When this parameter is transferred the RPS Pulse is equal to:

(RPS Pulse Width Extension value + 1) * (RPS Pulse Width value).

At power-on Reset or Selective Reset (Reset Slave), the RPS Pulse Width parameter value is defaulted to the number of octets per sector defined by the manufacturer's default specification or the last saved Format Specification if implemented and shall default to the new octets per sector if a Load Format Specification is accepted by the slave.

7.1.6.4 RPS Pulse Width Skew (Format 1 option)

This four-octet parameter may be transferred in addition to the previous octets to define the RPS Pulse Width Skew in octets. This parameter shall define a skew factor for the trailing edge of the RPS Pulse ahead of the trailing edge of the physical sector specified by the RPS Target Sector Address parameter. This skew factor should typically correspond to the master's estimated reconnect time in octets. In the case of large sectors, this allows the master to set the RPS Pulse immediately ahead of the desired sector and to define the number of octets prior to the desired sector the slave shall negate the RPS Pulse. When this parameter is used the master shall define the RPS pulse width in octets.

7.1.7 Load Position (07)

The Load Position command control causes the slave to seek to the physical cylinder specified by the Cylinder Address parameter, select the physical head specified by the Head Address parameter and be prepared to generate the RPS Pulse if the RPS Interrupt is enabled. RPS Interrupts, if enabled, are not asserted until the slave is on cylinder. Any active head offset or data strobe is cleared.

Unlike other commands, the Slave Ending Status of this command shall complete with the Time-Dependent Operation bit set. The Command Completion Interrupt is not asserted if a valid RPS Target Sector Address is accepted (which shall cause assertion of an RPS Interrupt).

There are several optional parameters which may be supported by the slave with this command. The slave shall terminate after the last parameter it supports. The master shall be prepared to master-terminate after the last parameter it transfers.

If an optional parameter is not supplied by the master, the value shall remain at its previous setting.

The format of the command parameters shall be as follows:

Octet	Parameter
0-3	Cylinder Address
4-5	Head Address
6-7	RPS Target Sector Address(Optional)
8-9	RPS Pulse Width Extension(Optional)
A-D	RPS Pulse Width(Optional)
E-11	RPS Pulse Width Skew(Optional)

See 7.1.6.1 - 7.1.6.4 for description of optional parameters.

7.1.8 Read Configuration (41)

The Read Configuration response control causes the slave to transfer configuration information to the master. The format for this transfer shall be as follows:

Octet	Bit	Parameters
0-1		Number of octets following: equals n - 1
2		Device Class Code 1 = disk
3		Slave type flag octet
	7	Non-removable disk
	6	Removable disk
	5	Reserved, set to zero
	4	Fixed head disk
	3	Moving head disk
	2	Solid State Disk
	1	Reserved, set to zero
	0	Reserved, set to zero
4		Capability flag octet
	7	Reserved, set to zero
	6	Variable Block Mode
	5	Fixed Block Mode, Sector Mode 2
	4	Fixed Block Mode, Sector Mode 1
	3	Soft sector
	2	Hard sector
	1	Field data controls
	0	Sector data controls
5		Feature flag octet
	7	RPS Target Sector Address
	6	Dual port

	5	Slave ECC
	4	Reserved, set to zero
	3	Slave responds to adjacent odd-even select addresses for dual actuators
	2	Slave responds to adjacent odd-even select addresses for dual devices per actuator
	1	Slave restores last loaded Format Specification
	0	Fixed Format Specification
6-9		Address of last data cylinder
A-D		Address of last defect list cylinder
E-F		Number of heads per cylinder
10-11		Number of fixed sectors per track
12-15		Number of octets per track
16-19		Single cylinder seek time (microseconds)
1A-1D		Average seek time (microseconds)
1E-21		Maximum seek time (microseconds)
22-25		Nominal rotation time (microseconds)
26-29		Head switching time (microseconds)
2A-2D		Write-to-read recovery time (microseconds)
2E-31		Manufacturer identification (ISO 646/ASCII)
32-39		Manufacturer model number (ISO 646/ASCII)
3A-3D		Manufacturer revision number (ISO 646/ASCII)
3E-45		Manufacturer unique unit ID number
46-47		Manufacturer switch settings
48-n		Manufacturer defined parameter

7.1.8.1 Number of Octets

This double-octet parameter contains the octet count for the Configuration Response. It does not include itself in the octet count.

7.1.8.2 Device Class Code

This single-octet parameter contains the device class code. This part of ISO/IEC 9318 defines only a single code, which is for magnetic disk drives.

7.1.8.3 Slave Type Flag Code

This single-octet parameter contains the slave type flag code. This parameter indicates whether the slave contains removable media, nonremovable media, or both; and has fixed heads, moving heads, or both.

7.1.8.4 Capability Flag Octet

This single-octet parameter contains the slave capability flag. This parameter indicates whether the slave can operate in the Variable Block Mode, the Fixed Block Mode, or in both modes. If the slave functions in the Fixed Block Mode, then this parameter indicates whether the slave supports:

a) Sector Mode 1, Sector Mode 2, or both (see 6.1.1)
b) Hard Sectors (see 6.1.1.1), Soft Sectors (see 6.1.1.2), or both
c) Field data controls, Sector data controls, or both (see 7.2.1.1)

7.1.8.5 Feature Flag Octet

This single-octet parameter contains the slave feature flag. This parameter indicates if the slave supports RPS Target Sector Address, Dual Port, Slave ECC, and adjacent odd-even select addresses for dual actuators.

7.1.8.6 Address of Last Data Cylinder

This four-octet parameter contains the address of the last data cylinder on this slave. The first cylinder has an address of zero.

7.1.8.7 Address of Last Defect List Cylinder

This four-octet parameter contains the address of the last defect list data cylinder on this slave.

7.1.8.8 Number of Heads Per Cylinder

This double-octet parameter contains the number of heads on this slave.

7.1.8.9 Number of Fixed Sectors per Track

This double-octet parameter contains the number of fixed sectors per track on this slave. If the slave is not fixed sectored, this parameter shall contain X'FFFF'.

7.1.8.10 Number of Octets per Track

This four-octet parameter contains the guaranteed number of physical octets per track on this slave.

7.1.8.11 Single-Cylinder Seek Time

This four-octet parameter contains the value of the single-cylinder seek time in microseconds for this slave.

7.1.8.12 Average Seek Time

This four-octet parameter contains the value of the average seek time in microseconds for this slave. Average seek time equals the sum of the times to perform all possible seeks divided by the number of all possible seeks.

7.1.8.13 Maximum Seek Time

This four-octet parameter contains the value of the maximum cylinder seek time in microseconds for this slave.

7.1.8.14 Rotational Time

This four-octet parameter contains the value of the nominal rotational time in microseconds for this slave.

7.1.8.15 Head Switching Time

This four-octet parameter contains the value of the maximum head switching time in microseconds for this slave.

7.1.8.16 Write-to-Read Recovery Time

This four-octet parameter contains the value of the maximum write-to-read recovery time in microseconds for this slave.

7.1.8.17 Manufacturer Identification

This four-octet parameter contains the slave manufacturer's Identification in ISO 646 (ASCII).

7.1.8.18 Manufacturer Model Number

This eight-octet parameter contains the slave model number in ISO 646 (ASCII).

7.1.8.19 Manufacturer Revision Number

This four-octet parameter contains the slave revision number in ISO 646 (ASCII).

7.1.8.20 Manufacturer Unit ID Number

This eight-octet parameter contains the unit identification number for the slave. The ID number for each slave, which is provided by the manufacturer, shall either be different from the numbers for any other slave of the same manufacturer, model, and revision number, or contain the value X'F...F', if not unique.

7.1.8.21 Manufacturer Switch Settings

This double-octet parameter contains the value of any switch settings the slave reports via the Configuration Response.

7.1.8.22 Manufacturer Defined Parameter

This optional parameter contains the manufacturer's defined values.

7.1.9 Read Format Specification (42)

The Read Format Specification response control causes the slave to transfer the current Format Specification to the master (see 6.2 for a description of the Format Specification). The response shall have the following format:

Octet	Parameters
0-1	Number of octets following: equals n - 1
2-3	Format Type and Flag Octet
4-n	Remainder of Format Specification

7.1.10 Read Slave Specific Information (43)

The Read Slave Specific Information response control causes the slave to transfer slave-specific information (e.g., diagnostics) to the master. The response shall have the following format:

Octet	Parameters
0-1	Number of octets following: equals n - 1
2-n	Slave-specific information

7.1.11 Read Status (44)

The Read Status response control causes the slave to transfer up to eight octets of status to the master. The response shall have the following format:

Octet	Parameters
0	Exception Status
1	Unsolicited Exception Status
2	Bus Control Exception Status
3-4	Slave Exception Status
5-7	Vendor Unique Status

See 8.2 for detailed definition of the Status Response. The slave may terminate the transfer at any point if the remaining octets are zeros.

Reading the status causes it to be cleared if the Master Status sent to the slave indicates a successful transfer.

7.1.12 Read Correction Vectors (45) (optional)

The Read Correction Vectors response control causes the slave to transfer the ECC correction vectors to the master. If the optional slave ECC is not implemented, this control shall be rejected as unsupported. The format of the response shall be as follows:

Octet	Parameters
0-1	Number of octets following: equals n - 1
2	Error pattern
3-5	Error octet location from start of last field transferred
6-n	Octets 2-5 repeated as needed for additional vectors

The error pattern octet is exclusive-ORed with the data octet at the specified location. If an error is in the ECC octets, no vectors are returned.

7.1.13 Read Current Sector Address (46)

The Read Current Sector Address response control causes the slave to transfer the current physical sector address to the master. The response shall have the following format:

Octet	Parameter
0-1	Current Sector Address

If the slave cannot determine the Current Sector Address, the Current Sector Address is X'FFFF'. The Current Sector Address received by the master may differ from actual physical sector location because of the time required to transfer the response.

7.1.14 Read Current Position (47)

The Read Current Position response control causes the slave to transfer the current position to the master. The format of the response shall be as follows:

Octet	Parameters
0-3	Current Cylinder Address
4-5	Current Head Address
6-7	Current RPS Target Sector Address
8-9	Current Sector Address

If the RPS Interrupt is disabled, the Current RPS Target Sector Address shall be set to X'FFFF'.

If the RPS option is not supported, the Current RPS Target Sector Address shall be set to X'0000'.

If the slave cannot determine the Current Sector Address, the Current Sector Address shall be set to X'FFFF'. The Current Sector Address received by the master may differ from the actual physical sector because of the time required to transfer the response.

7.1.15 Read Extended Status (48)

The Read Extended Status response control causes the slave to transfer up to eight octets of status to the master. The format of the response shall be as follows:

Octet	Parameters
0	Interface Flags
1	Data Recovery Flags
2	Slave Control Flags

3 Slave Status
4 Slave Alarms
5-7 Vendor Unique Status

See 8.3 for detailed definition of the Extended Status Response. The slave may terminate the transfer at any point if the remaining octets are zeros.

7.2 Fixed Block Data Controls

Fixed Block data controls enable the users to read and write on the disk in the Fixed Block Format. They specify the direction of the transfer, the fields to be involved, the orientation of the fields, and Head Advance control.

7.2.1 Definitions and use

7.2.1.1 Fixed Block data control types

There are two types of Fixed Block data controls: Field data controls and Sector data controls. Field data controls specify an operation on a single field or a pair of fields. Sector data controls are composite controls that specify operations on sectors having a header and data fields as required.

When using Field data controls, the previous field shall be operated on by a Field or Sector data control.

7.2.1.2 Head Advance Control

When bit 4 is set in any data control, the head counter advances at the end of a successful transfer. It is also advanced unconditionally by the Step Head data control. (see 6.6)

7.2.1.3 Orientation with the disk

It is essential that the master's issuing of data controls stays oriented with the disk and that no sectors or fields, which are intended to be operated on, are missed. A data control that is not received in time for the slave to act on the next sector or field following the previous data control causes orientation to be lost and is rejected. To prevent the first data control of a series from being rejected when there is no orientation, orientation shall be established by starting with a target-type sector data control or issuing a read or verify header-type sector data control.

A field data control that operates on the header field shall set the sector orientation for that sector.

Orientation is not lost by a header verify miscompare or a master-initiated termination of a transfer.

Orientation shall be lost on a Head Advance data control if during the head advance one or more sectors pass under the head prior to the head settling.

7.2.1.4 Target Sector

A group of read and write data controls operate on the Target Sector.

If the RPS Target Sector Address option is supported, the target is the sector identified by the master in a Load RPS Target Sector Address or Load Position command control.

If the RPS Target Sector Address option is not supported the RPS Target Sector Address shall be X'0000'.

7.2.1.5 Sector data control reject

Sector data controls may be rejected for several reasons, including:

- The Format Specification is not initialized;
- The slave is active on the alternate port;
- The sector data control is a write type and the slave is write protected;
- The orientation is lost;

- The data control type is invalid.

7.2.1.6 Fixed Block data control coding

The Fixed Block data control, which is a form of the Bus Control, has the following coding:

Bit	Description
7	1 = Sector data control 0 = Command/Response control
6	1 = Information In (Read) 0 = Information Out (Write)
5	0 = Fixed Block sector data control
4	Step Head control
3	Read/Write Header Field operation
2	Modifier Bit Selects Target Sector Address for Read/Write operations Selects Verify Header for write operation Selects Skip Header for read operation
1	Data Field 2 operation
0	Data Field 1 operation

This coding results in eight groupings of Data Controls plus a special control as shown in the following table:

Bus Control Codes	Data Control
80-83 91-93	Skip/Write Data Field
84-87 94-97	Verify Header, Write Data
88-8B 98-9B	Write Header, Write Data
8C-8F 9C-9F	Write Header, Write Data At Target Sector Address
C0-C3 D1-D3	Skip/Read Data Field
C4-C7 D4-D7	Skip Header, Read Data
C8-CB D8-DB	Read Header, Read Data
CC-CF DC-DF	Read Header, Read Data At Target Sector Address
90	Step Head
D0	reserved

7.2.2 Sector Data Controls (optional)

Sector data controls operate on sectors consisting of a Header Field and one or more optional Data Fields. These data controls are described in 7.2.2.1 through 7.2.2.24. The first of the two hexadecimal codes is for no head advance; the second is for head advance.

No data is transferred during a Skip operation.

7.2.2.1 Verify Header (84, 94)

The Verify Header sector data control causes the slave to verify the next Header Field against data transferred from the master. If the header verifies, the slave shall indicate a Successful Information Transfer in the Slave Status. If the header does not verify, the slave shall terminate the transfer after the header and indicate Verify Header Miscompare in the Slave Status.

7.2.2.2 Verify Header and Write Data Field 1 (85, 95)

The Verify Header and Write Data Field 1 sector data control causes the slave to verify the next Header Field against data transferred from the master. If the header verifies, the slave continues to write Data Field 1 with data transferred from the master. If the header does not verify, the slave shall terminate the transfer after the header and indicate Verify Header Miscompare in the Slave Status.

7.2.2.3 Verify Header and Write Data Field 2 (86, 96)

The Verify Header and Write Data Field 2 sector data control causes the slave to verify the next Header Field against data transferred from the master. If the header verifies, the slave continues to write Data Field 2 with data transferred from the master. If the header does not verify, the slave shall terminate the transfer after the header and indicate Verify Header Miscompare in the Slave Status.

7.2.2.4 Verify Header and Write Data Fields 1 and 2 (87, 97)

The Verify Header and Write Data Fields 1 and 2 sector data control causes the slave to verify the next Header Field against data transferred from the master. If the header verifies, the slave continues to write Data Fields 1 and 2 with data transferred from the master. If the header does not verify, the slave shall terminate the transfer after the header and indicate Verify Header Miscompare in the Slave Status.

7.2.2.5 Write Header (88, 98)

The Write Header sector data control causes the slave to write the next Header Field with data transferred from the master.

7.2.2.6 Write Header and Data Field 1 (89, 99)

The Write Header and Data Field 1 sector data control causes the slave to write the next Header Field and Data Field 1 with data transferred from the master.

7.2.2.7 Write Header and Data Field 2 (8A, 9A)

The Write Header and Data Field 2 sector data control causes the slave to write the next Header Field and Data Field 2 with data transferred from the master.

7.2.2.8 Write Header and Data Fields 1 and 2 (8B, 9B)

The Write Header and Data Fields 1 and 2 sector data control causes the slave to write the next Header Field and Data Fields 1 and 2 with data transferred from the master.

7.2.2.9 Write Header at Target Sector Address (8C, 9C)

The Write Header At Target sector data control causes the slave to write the Header Field at the target sector with data transferred from the master.

7.2.2.10 Write Header and Data Field 1 at target (8D, 9D)

The Write Header and Data Field 1 At Target sector data control causes the slave to write the Header Field and Data Field 1 at the target sector with data transferred from the master.

7.2.2.11 Write Header and Data Field 2 at Target (8E, 9E)

The Write Header and Data Field 2 At Target sector data control causes the slave to write the Header Field and Data Field 2 at the target sector with data transferred from the master.

7.2.2.12 Write Header and Data Fields 1 and 2 at Target (8F, 9F)

The Write Header and Data Fields 1 and 2 At Target sector data control causes the slave to write the Header Field and Data Fields 1 and 2 at the target sector with data transferred from the master.

7.2.2.13 Skip Header (C4, D4)

The Skip Header sector data controls causes the slave to skip the next Header Field. No data is transferred and the slave initiates the Ending Status sequence at the end of the Header Field.

7.2.2.14 Skip Header and Read Data Field 1 (C5, D5)

The Skip Header and Read Data Field 1 sector data control causes the slave to skip the next Header Field and transfer Data Field 1 to the master.

7.2.2.15 Skip Header and Read Data Field 2 (C6, D6)

The Skip Header and Read Data Field 2 sector data control causes the slave to skip the next Header Field and transfer Data Field 2 to the master.

7.2.2.16 Skip Header and Read Data Fields 1 and 2 (C7, D7)

The Skip Header and Read Data Fields 1 and 2 sector data control causes the slave to skip the next Header Field and transfer Data Fields 1 and 2 to the master.

7.2.2.17 Read Header (C8, D8)

The Read Header sector data control causes the slave to transfer the next Header Field to the master.

7.2.2.18 Read Header and Data Field 1 (C9, D9)

The Read Header and Data Field 1 sector data control causes the slave to transfer the next Header Field and Data Field 1 to the master.

7.2.2.19 Read Header and Data Field 2 (CA, DA)

The Read Header and Data Field 2 sector data control causes the slave to transfer the next Header Field and Data Field 2 to the master.

7.2.2.20 Read Header and Data Fields 1 and 2 (CB, DB)

The Read Header and Data Fields 1 and 2 sector data control causes the slave to transfer the next Header Field and Data Fields 1 and 2 to the master.

7.2.2.21 Read Header at Target (CC, DC)

The Read Header At Target sector data control causes the slave to transfer the Header Field at the target sector to the master.

7.2.2.22 Read Header and Data Field 1 at Target (CD, DD)

The Read Header and Data Field 1 At Target sector data control causes the slave to transfer the Header Field and Data Field 1 at the target sector to the master.

7.2.2.23 Read Header and Data Field 2 at Target (CE, DE)

The Read Header and Data Field 2 At Target sector data control causes the slave to transfer the Header Field and Data Field 2 at the target sector to the master.

7.2.2.24 Read Header and Data Fields 1 and 2 at Target (CF, DF)

The Read Header and Data Field 1 and 2 At Target sector data control causes the slave to transfer the Header Field and Data Fields 1 and 2 at the target sector to the master.

7.2.3 Field Data Controls (optional)

Field data controls operate on a single field or a pair of fields and are described in 7.2.3.1 through 7.2.3.8.

No data is transferred during a Skip operation.

The first hexadecimal code is for no head advance; the second is for head advance.

7.2.3.1 Skip Field (80)

The Skip Field data control causes the slave to skip the next field.

7.2.3.2 Skip Two Fields (C0)

The Skip Two Fields data control causes the slave to skip the next two fields.

7.2.3.3 Write Field (81, 91)

The Write Field data control causes the slave to write the next field with data transferred from the master.

7.2.3.4 Skip Field and Write Field (82, 92)

The Skip Field and Write Field data control causes the slave to skip the next field and then write the following field with data transferred from the master.

7.2.3.5 Write Two Fields (83, 93)

The Write Two Fields data control causes the slave to write the next two fields with data transferred from the master.

7.2.3.6 Read Field (C1, D1)

The Read Field data control causes the slave to transfer the next field to the master.

7.2.3.7 Skip Field and Read Field (C2, D2)

The Skip Field and Read Field data control causes the slave to skip the next field and then transfer the following field to the master.

7.2.3.8 Read Two Fields (C3, D3)

The Read Two Fields data control causes the slave to transfer the next two fields to the master.

7.2.4 Special data controls - Step Head (90)

The Step Head data control causes the slave to advance to the next head address. No data is transferred. (see 6.6)

7.3 Variable Block Data Controls (optional)

The Variable Block data controls provide for reading and writing Data Fields of variable length on the disk. They specify the direction of the data transfer, the field type involved, and the orientation requirements of the field.

7.3.1 Variable block data control coding

A Variable Block data control, which is a form of the Bus Control, has the following coding:

Bit	Description
7	1 = Data control, 0 = Command/Response Control
6	1 = Information In (Read), 0 = Information Out (Write)
5	1 = Variable Block Control
4-0	x = Read/Write Variable Block Data Operation Code

The variable block operations shall be defined as follows:

Bus Control Code	Description
A0-A3	Format Field type
A4-A7	Format Field type
A8-AB	Write Field type
AC	Skip Field type
AD-AF	reserved
90	Step Head(See 7.2.4)
B0-BF	reserved
E0-E3	Read Field Type
E4-E7	reserved
E8-EB	reserved
EC-EF	reserved
F0-FF	reserved

7.3.2 Format Home Field (A0)

This data control causes the slave to write the Home Field following Index Mark and to write a sync pattern until the index is encountered a second time.

This data control does not require orientation and can be issued at any time by the master. The slave locates the Home Field by reference to the Format Specification, which defines the distance in cells between the beginning of the Home Field and the disk Index Mark.

The slave shall terminate the data transfer when the end of the Home Field data segment is encountered, as defined in the Format Specification. The slave shall then begin writing a Home Gap. If another format-type data control is received before the Home Gap is completely written, it shall be executed at the end of the Home Gap. Otherwise, a sync pattern shall be written until Index Mark is detected.

7.3.3 Format Header Field (A1)

This data control causes the slave to format a Header Field. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, Skip Field or another Data Field. When the end of the current field gap is reached, the slave writes an Address Mark, Sync Pattern, and Header data segment.

The slave shall terminate the data transfer when the end of the Header Field Data Segment is reached, as defined in the Format Specification. The slave shall then begin writing a Header Gap. If another format-type Control is received before

the Header Gap is completely written, it shall be executed at the end of the Header Gap. Otherwise, a sync pattern shall be written until Index Mark is detected.

7.3.4 Format Data Field (A2)

This data control causes the slave to write the next Data Field followed by a Data Field Gap Segment. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, Skip Field, or another Data Field. The slave locates the Data Field by assuming it immediately follows the current field.

The end of the operation occurs when the master terminates the data transfer. The slave shall then begin writing a Data Field Gap. If another format-type Control is received before the Data Field Gap is completely written, it shall be executed at the end of the Data Field Gap. Otherwise, a sync pattern shall be written until Index Mark is detected.

7.3.5 Format Data Field — Short Skip (A3)

This data control causes the slave to write the next Data Field followed by a Data Field Skip Segment. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, Skip Field, or another Data Field. The slave locates the Data Field by assuming it immediately follows the current field.

The end of the operation occurs when the master terminates the data transfer. The slave shall then begin writing a Data Field Skip. If another format-type data control is received before the Data Field Skip is completely written, it shall be executed. Otherwise, a sync pattern shall be written until Index Mark is detected.

7.3.6 Format Skip from Index (A4)

This data control causes the slave to write a sync pattern from index. The length of the sync gap in cells is specified by the double-octet parameter passed to the slave. This data control does not require orientation and can be issued at any time by the master.

The slave shall terminate the data transfer after receiving the gap length parameter. The slave shall begin writing an Index Gap after detecting the Index Mark. If another format-type data control is received before the Index Gap is completely written, it shall be executed at the end of the Index Gap.

Otherwise, a sync pattern shall be written until Index Mark is detected again.

7.3.7 Format Skip (A5)

This data control causes the slave to write a sync pattern from the end of the gap of the previous format-type data control. The length of the sync gap in cells is specified by the double-octet parameter passed to the slave.

This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, Skip Field or Data Field.

The slave shall terminate the data transfer after receiving the gap length parameter. The slave shall begin writing a Format Gap after detecting the end of the current field gap. If another format-type data control is received before the Format Gap is completely written, it shall be executed at the end of the Format Gap. Otherwise, a sync pattern shall be written until Index Mark is detected again.

7.3.8 Write Home Field (A8)

The Write Home Field data control causes the slave to write the Home Field following index. This data control does not require orientation and can be issued at any time by the Master. The slave locates the Home Field by reference to the Format Specification, which defines the distance, in cells between the beginning of the Home Field and disk Index Mark. The end of the operation occurs when the slave terminates the data transfer.

The slave shall terminate the data transfer when the end of the Home Field data segment is encountered, as defined in the Format Specification. If a Data Field follows the Home Field and the master intends to operate on it, the appropriate Field control shall be issued before the end of the Home Field Gap is reached.

7.3.9 Write Header Field (A9)

The Write Header Field data control causes the slave to write in the next Header Field encountered. This data control does not require orientation and can be issued at any time by the master. The slave locates the Header Field by searching for the first occurrence of an Address Mark after receipt of this data control. The end of the operation occurs when the slave terminates the data transfer.

The slave shall terminate the data transfer when the end of the Header Field data segment is encountered, as defined in the Format Specification.

If any Data Fields follow the Header Field and the master intends to operate on them, the appropriate Field control shall be issued before the end of the Header Field Gap is reached.

7.3.10 Write Data Field (AA)

The Write Data Field data control causes the slave to write the next Data Field. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, Skip Field, or another data field. The slave locates the Data Field by assuming it immediately follows the current field. The end of the operation occurs when the master terminates the data transfer. After Master Termination the slave shall continue to maintain orientation until the end of the Data Field Gap Segment.

7.3.11 Write Data Field - Short Skip (AB)

This data control causes the slave to write the next Data Field. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, or another Data Field. The slave locates the Data Field by assuming it immediately follows the current field.

7.3.12 Defect Skip (AC)

This data control causes the slave to skip from the end of the last field gap, by the number of cells specified by the double-octet parameter passed to the slave as write data. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, or a Data Field.

7.3.13 Read Home Field (E0)

The Read Home Field data control causes the slave to read the Home Field following index. This data control does not require orientation and can be issued at any time by the master. The slave locates the Home Field by reference to the Format Specification, which defines the distance in cells between the beginning of the Home Field and the Disk Index Mark. The end of the operation occurs when either the master or the slave terminates the data transfer.

The master may terminate the data transfer operation at any time, and the slave shall continue to maintain orientation until the end of field is reached, as defined in the Format Specification.

The slave shall terminate the data transfer when the end of the Home Field data segment is encountered, as defined in the Format Specification. If a Header or Data Field follows the Home Field and the master intends to operate on it, the appropriate Field Control shall be issued before the end of the Home Field Gap is reached.

7.3.14 Read Header Field (E1)

The Read Header Field data control causes the slave to read the next Header Field encountered. This data control does not require orientation and can be issued at any time by the master. The slave locates the Header Field by searching for the first occurrence of a Header Address Mark after receipt of this data control. The end of the operation occurs when either the master or the slave terminates the data transfer.

The master may terminate the data transfer operation at any time, and the slave shall continue to maintain orientation until the end of the field is reached, as defined in the Format Specification.

The slave shall terminate the data transfer when the end of the Header Field data segment is encountered after receipt of this data control. The end of the operation occurs when the slave terminates the data transfer.

The slave shall terminate the data transfer when the end of the Header Field data segment is encountered, as defined in the Format Specification.

If any Data Fields follow the Header Field and the master intends to operate on them, the appropriate Field control shall be issued before the end of the Header Field Gap is reached.

7.3.15 Read Data Field (E2)

The Read Data Field data control causes the slave to read the next Data Field. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, or another Data Field. The slave locates the Data Field by assuming it immediately follows the current field. The end of the operation occurs when the master terminates the data transfer.

The master may terminate the data transfer operation at any time before the end of the field is reached. However, if any Data Fields follow this one and the master intends to operate on them, termination of the transfer shall occur at the end of the Data Fields, since the slave uses Master Termination to determine the end of the Data Segment and the beginning of the Field Gap.

After Master Termination, the slave shall continue to maintain orientation until the end of Field Gap is reached, as defined in the Format Specification.

7.3.16 Read Data Field — Short Skip (E3)

The Read Data Field data control causes the slave to read the next data field. This data control requires orientation and can only be issued during the gap following a Home Field, Header Field, or another Data Field. The slave locates the Data Field by assuming it immediately follows the current field. The end of the operation occurs when the master terminates the data transfer.

The master may terminate the data transfer operation at any time before the end of the field is reached. However, if any Data Fields follow this one and the master intends to operate on them, termination of the transfer shall occur at the end of Data Fields, since the slave uses Master Termination to determine the end of the Data Segment and the beginning of the Short Skip Field Gap. After Master Termination, the slave shall continue to maintain orientation until the end of the Short Skip Field Gap is reached, as defined in the Format Specification.

8 Status

8.1 Slave Status

The Slave Status Octet is presented to the master during the Ending Status sequence following an Information Transfer and is defined as follows:

Bit	Description
7	Successful Information Transfer
6	Bus Parity Error
5	Odd Octet Transfer
4	Time-Dependent Operation
3-0	Operation Ending Status

00 xx = Normal end
- 00 = Slave available, Bus Control executed
- 01 = Slave Busy, Bus Control rejected
- 1x = reserved

01 xx = Data Exceptions
- 00 = missed AM
- 01 = missed Sync Byte
- 10 = ECC error (optional)
- 11 = Verify Header Miscompare

10 00 = Operation Exceptions
10 01 = reserved
10 1x = reserved
11 00 = Unsolicited Exceptions
11 01 = reserved
11 1x = reserved

8.1.1 Successful Information Transfer

This bit shall be set if the slave determines that the Information Transfer was successful.

8.1.2 Bus Parity Error

This bit shall be set if there is a parity error on the Bus Control Octet, Master Status Octet or any information received by the slave.

8.1.3 Odd Octet Transfer

This bit shall be set if the data transfer was of an odd octet length. The octet on Bus "B" of the last word transferred should be ignored when this bit is true.

8.1.4 Time-Dependent Operation

This bit shall be set if the command has not been completed by the slave at the time this status is sent to the master. When the command has completed one of the following shall occur:

- Command Completion Interrupt shall be asserted;
- RPS Pulse shall be asserted. (See 7.1.7 - Load Position);
- Status Pending Interrupt shall be asserted. This shall happen if an exception occurs before the command completes.

8.1.5 Operation Ending Status

This four-bit field indicates that the attempted Bus Control was rejected because of a slave busy condition or outstanding Unsolicited exception, or if the preceding Bus Control incurred a Data or Operation exception. A slave busy condition can result because the slave is reserved by the alternate port or because the slave is currently executing a command that previously returned an Ending Status Octet with the Time-Dependent Operation (TDO) bit set. There are three types of categories of exceptions reported in the Ending Status Octet: Data Exceptions, Operation Exceptions, and Unsolicited Exceptions.

8.1.5.1 Data Exceptions

These exceptions occur when there is a failure to Read or Write data successfully.

8.1.5.1.1 Missed AM

This exception shows a failure to detect Address Mark.

8.1.5.1.2 Missed Sync Byte

This exception shows a failure to detect the Sync Byte field.

8.1.5.1.3 ECC Error (optional)

This exception shows that an ECC error was detected.

8.1.5.1.4 Verify Header Miscompare

This exception occurs when there is a mismatch between header and master supplied data.

8.1.5.2 Unsolicited Exception

An Unsolicited Exception indicates that the attempted Bus Control was rejected because outstanding Unsolicited Exception Status exists and the master shall read this status by issuing a Read Status Response Command before any other Bus Controls shall be accepted.

8.1.5.3 Operation Exception

An Operation Exception indicates that the attempted Bus Control incurred an operation error, and that the nature of this error may be determined by issuing a Read Status Response Command. This captured status may be ignored.

If the available status is ignored, it shall be clear upon receipt of the next Bus Control.

8.2 Status Response

The status bits of the Status Response indicate exception conditions. They are set on the occurrence of an exception event. The setting of any exception status bit activates the Status Pending interrupt and ATTENTION IN (if enabled), unless previously reported at Slave Ending Status.

The slave transfers the Status Response to the master upon receiving a Read Status response control and clears all exception status bits if the Master Status Octet indicates a successful transfer. The exception status bits are also cleared by a Reset Logical Interface. All exception status bits, except the Unsolicited Exception bits (octet 0, bit 6 and octet 1, bits 7 through 0) are also cleared upon the acceptance of any Bus Control.

The Status Response shall have the following format:

Octet	Bit	Description
0		Exception Status Octet
	7	Status Response
	6	Unsolicited Exception
	5	Bus Control Exception
	4	Read Fault
	3	Write Fault
	2	Seek Fault
	1	Spindle Fault
	0	Execution Fault
1		Unsolicited Exception Status
	7	Reset Complete
	6	Alternate Port Priority Select
	5	Alternate Port Format Change
	4	Alternate Port Format Complete
	3	reserved, set to zero
	2	Not Ready Transition
	1	Ready Transition
	0	Media Change
2		Bus Control Exception Status
	7	Invalid Bus Control
	6	Invalid Parameter
	5	Unsupported Bus Control or Function Code
	4	Bus Control or Function Code Context
	3	Data Bus Control Late
	2	reserved, set to zero
	1	reserved, set to zero
	0	reserved, set to zero
3		Slave Exception Status
	7	Speed Fault
	6	Off Cylinder Fault
	5	Head Select Fault
	4	reserved, set to zero
	3	reserved, set to zero
	2	Voltage Fault
	1	Logic Temperature Fault
	0	Actuator Temperature Fault
4		Slave Exception Status
	7	Write Protect Fault
	6	Write Current Fault
	5	Write Transition Fault
	4	Head Offset Fault
	3	Data Strobe Fault
	2	reserved, set to zero
	1	reserved, set to zero
	0	reserved, set to zero
5	7-0	Vendor Unique Status
6	7-0	Vendor Unique Status
7	7-0	Vendor Unique Status

8.2.1 Exception Status Octet (octet 0)

8.2.1.1 Status Response (octet 0 bit 7)

This bit is always zero for a Status Response.

8.2.1.2 Unsolicited Exception (octet 0 bit 6)

This bit shall be set when the slave has incurred an Unsolicited Exception condition. At least one bit shall be set in octet 1, describing the exception type. This bit and the bits in octet 1 can only be reset (on a port basis) by issuing a Read Status response control followed by a Master Status Octet with the Successful Information Transfer bit set. As long as an Unsolicited Exception condition exists for a port, the slave shall reject all Bus Controls from that port and only that port, except Read Status Response, with the Unsolicited Exception bit set (bit 3) in the Slave Status Octet.

8.2.1.3 Bus Control Exception (octet 0 bit 5)

This bit shall be set when the slave rejects the Bus Control as invalid, since it contains a parameter that is invalid, unsupported, out of context, or late. At least one bit shall be set in octet 2 to describe the type of command exception.

8.2.1.4 Read Fault (octet 0 bit 4)

This bit shall be set when the slave detects an error while executing a Read/Verify Data Control. At least one bit shall be set in octets 3 through 7 to describe the type of read exception.

8.2.1.5 Write Fault (octet 0 bit 3)

This bit shall be set when the slave detects an error while executing a Write Data Control. At least one bit shall be set in octets 3 through 7 to describe the type of write exception.

8.2.1.6 Seek Fault (octet 0 bit 2)

This bit shall be set when the slave detects a seek error while executing a Seek Command, Read/Verify Data, or Write data control. At least one bit shall be set in octets 3 through 7 to describe the type of seek exception.

8.2.1.7 Spindle Fault (octet 0 bit 1)

This bit shall be set when the slave detects a spindle error while executing a Spin Up or Spin Down function code, Read/Verify Data, or Write Data Control. At least one bit shall be set in octets 3 through 7 to describe the type of spindle exception.

8.2.1.8 Execution Fault (octet 0 bit 0)

This bit shall be set when the slave detects an Execution error other than Read, Write, Seek, or Spindle Fault during the execution of a Command or Data Control. At least one bit shall be set in octets 3 through 7 to describe the type of exception.

8.2.2 Unsolicited Exception Status (octet 1)

8.2.2.1 Reset Complete (octet 1 bit 7)

This bit shall be set when the slave has executed and completed a Reset Slave (see 6.5.1).

8.2.2.2 Alternate Port Priority Select (octet 1 bit 6)

This bit shall be set when the alternate port receives a Selection Octet with the Priority Select bit set.

8.2.2.3 Alternate Port Format Change (octet 1 bit 5)

This bit shall be set when the alternate port accepts a Format Specification.

8.2.2.4 Alternate Port Format Complete (octet 1 bit 4)

This bit shall be set when the alternate port receives a Load Slave function code 16, Notify Alternate Port of Format Completion.

8.2.2.5 Reserved for future use (octet 1 bit 3)

8.2.2.6 Not Ready Transition (octet 1 bit 2)

This bit shall be set when the slave goes from a ready to a not ready condition.

8.2.2.7 Ready Transition (octet 1 bit 1)

This bit shall be set when the slave goes from a not ready to ready condition.

8.2.2.8 Media Change (octet 1 bit 0)

This bit shall be set with the Ready Transition bit (bit 1) if the slave detects that the media was removed and replaced previous to the Not Ready Transition to Ready Transition.

8.2.3 Bus Control Exception Status (octet 2)

8.2.3.1 Invalid Bus Control (octet 2 bit 7)

This bit shall be set when a Bus Control was received that is not defined in this International Standard.

8.2.3.2 Invalid Parameter (octet 2 bit 6)

This bit shall be set when a valid Bus Control was received with an invalid parameter.

8.2.3.3 Unsupported Bus Control or Function Code (octet 2 bit 5)

This bit shall be set when a valid Bus Control or valid Function Code is received that is not supported by this slave.

8.2.3.4 Bus Control or Function Code Context (octet 2 bit 4)

This bit shall be set when a valid Bus Control or valid Function Code is received but cannot be executed because it conflicts with the current context of the slave.

8.2.3.5 Data Control Late (octet 2 bit 3)

This bit shall be set when a valid Data Control is received later than the orientation window for the next field.

8.2.3.6 Reserved for future use (octet 2 bit 2)

8.2.3.7 Reserved for future use (octet 2 bit 1)

8.2.3.8 Reserved for future use (octet 2 bit 0)

8.2.4 Slave Exception Status (octet 3)

8.2.4.1 Speed Fault (octet 3 bit 7)

This bit shall be set if the slave did not reach the required speed in the required time during the execution of a Spin Up function code or loses speed during a Data Control.

8.2.4.2 Off Cylinder Fault (octet 3 bit 6)

This bit shall be set if the slave did not come on cylinder in the required time during the execution of a Seek, or loses"on cylinder" during a Data Control.

8.2.4.3 Head Select Fault (octet 3 bit 5)

This bit shall be set if the slave detects an invalid head selection.

8.2.4.4 Reserved for future use (octet 3 bit 4)

8.2.4.5 Reserved for future use (octet 3 bit 3)

8.2.4.6 Voltage Fault (octet 3 bit 2)

This bit shall be set if the slave detects a voltage out of range condition.

8.2.4.7 Logic Temperature Fault (octet 3 bit 1)

This bit shall be set if the slave detects an over temperature condition in the slave electronics.

8.2.4.8 Actuator Temperature Fault (octet 3 bit 0)

This bit shall be set if the slave detects an over temperature condition in the slave servo actuator.

8.2.5 Slave Exception Status (octet 4)

8.2.5.1 Write Protect Fault (octet 4 bit 7)

This bit shall be set when a Write data control is received and the slave is write protected.

8.2.5.2 Write Current Fault (octet 4 bit 6)

This bit shall be set if write current was out of range during the execution of a Write data control.

8.2.5.3 Write Transition Fault (octet 4 bit 5)

This bit shall be set if no write transitions are detected by the slave during the execution of a Write data control.

8.2.5.4 Head Offset Fault (octet 4 bit 4)

This bit shall be set if the data heads were in an offset position when a Write data control was received.

8.2.5.5 Data Strobe Fault (octet 4 bit 3)

This bit shall be set if the slave receives a Write data control when early or late data strobe is in effect.

8.2.5.6 Reserved for future use (octet 4 bit 2)

8.2.5.7 Reserved for future use (octet 4 bit 1)

8.2.5.8 Reserved for future use (octet 4 bit 0)

8.2.6 Vendor unique status (octet 5 bits 7-0)

8.2.7 Vendor unique status (octet 6 bits 7-0)

8.2.8 Vendor unique status (octet 7 bits 7-0)

8.3 Extended Status Response

The slave transfers Extended Status Response to the master upon receiving a Read Extended Status Bus Control. The Extended Status bits are static indications of current flag states and slave conditions.

The Extended Status Response shall have the following format:

Octet	Bit	Description
0		Interface Flags
	7	Extended Status
	6	Port Number
	5	Alternate Port Enabled
	4	Reserve Active
	3	Command Complete Interrupt Enabled
	2	RPS Interrupt Enabled
	1	Status Pending Interrupt Enabled
	0	Format Specification Present
1		Data Recovery Flags
	7	Offset Direction
	6	Offset MSB
	5	Offset LSB
	4	Early Strobe
	3	Late Strobe
	2	reserved, set to zero
	1	Header Field ECC/CRC Enable
	0	Data Field ECC/CRC Enable
2		Slave Control Flags
	7	Write Protected
	6	Spindle Power On
	5	Lock Carriage
	4	Load Heads
	3	reserved, set to zero
	2	reserved, set to zero
	1	reserved, set to zero
	0	reserved, set to zero
3		Slave Status
	7	Speed
	6	On Cylinder
	5	reserved, set to zero
	4	reserved, set to zero
	3	reserved, set to zero
	2	reserved, set to zero
	1	Head Disk Assembly (HDA) Ready
	0	Media Present
4		Slave Alarms
	7	reserved, set to zero
	6	reserved, set to zero
	5	Illegal Head Select
	4	reserved, set to zero
	3	reserved, set to zero
	2	Voltage Range Error
	1	Logic Over Temperature
	0	Actuator Over Temperature
5	7-0	Vendor Unique Extended Status
6	7-0	Vendor Unique Extended Status
7	7-0	Vendor Unique Extended Status