X3T9.2/92-027

Serial Storage Architecture
SSA-1 (Disk Order Level)
Version 1.0

12th December, 1991

IBM Corporation

This document is preliminary and the contents are subject to change without notice. Enquiries, suggestions and requests for additional copies may be directed to the following:

Primary contact:
Colin G. Butterworth,
IBM Corporation,
5600 Cottle Road,
San Jose,
California 95193,
U.S.A.
Tel: (408)-256-3671

Secondary contact:
William G. Verdoorn,
IBM Corporation,
Hwy. 52 and 37th. St. NW,
Rochester,
Minnesota 55901,
U.S.A.
Tel: (507)-253-2197

IBM may use any information that you supply without incurring any obligation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program or service is not intended to state or imply that only IBM's product, program or service may be used. Any functionally equivalent product, program or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

This publication may refer to products that are announced but not currently available in your country. This publication may also refer to products that have not been announced in your country. IBM makes no commitment to make available any unannounced products referred to herein, or any product employing methods or devices described herein. The final decision to announce any product is based on IBM's business and technical judgment.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Director of Commercial Relations,
IBM Corporation,
Purchase,
New York 10577,
U.S.A.

Contents

Introduction

This document describes the low level order set for disk drives in the Serial Storage Architecture (SSA) that is used on the serial link to the disk drive. SSA defines a hierarchy of serial interfaces for use within future storage sub-systems. It is divided into three layers, each described in a separate document:

SSA-0 A common transport layer used by SSA-1 and SSA-2.

SSA-1 A low-level disk drive order set.

SSA-2 A high-level (SCSI) command set.

The SSA-0 transport layer is quite general and it may be used for other purposes within its distance and performance capabilities. SSA-1 and SSA-2 are alternative 'upper-level protocols'.

At the time of writing, the initial level of SSA-0 has been written. SSA-2 is not yet available.

Overview of Disk Drive Functions

The prime use of the Serial Storage Architecture, level 1 (SSA-1) is to attach disk drives to controllers in clusters and arrays.

The objectives of the SSA-1 function split between the controller and the device attached on the serial link are:

1. Ease of use

   There are no device dependent functions in the controller.

   The number of optional orders and options within orders is limited to allow different devices to be attached to a controller with minimum integration.

2. Minimize overheads by, for example, avoiding duplication of controller functions such as command queuing and caching.

3. Allow the implementation to determine the amount of buffering

   In array environments, a single level of large buffering in the array controller is desirable to maximize performance. The function split between the controller and the disk drives assumes only a limited buffer in the drive. A disk drive may implement a larger buffer, but this need not be used for read ahead or cache data.

4. Allow tight device control

   To maximize performance, the controller has tight control on split read and write operations to allow data transfer between the using system and the media to start from the block that is first accessed rather than from the first block in the specified range. Sequential write operations can be executed back to back without a loss of a revolution. Read ahead to the controller can be quickly terminated.

5. Integrated spindle synchronization

   Spindle synchronization is supported on the serial link without requiring a separate cable for that specific purpose.

The principal functions of the disk drive attaching to the serial link are:

- It provides a serial interface to the controller. The controller sends orders to the disk drive in message frames. The disk drive sends status and requests for data in message frames. Data is transferred in data frames.

- Attachment to multiple links is allowed by the architecture. This is primarily for availability in case one controller fails.
- Read and write orders access multiple consecutive sectors of customer data. The disk drive automatically switches heads and if necessary seeks to the next cylinder to continue a read or write on the next track. The controller supplies the starting logical block address (LBA), block count and write data. Address translation to the physical cylinder and head required is performed in the disk drive.

  Bypassing defective areas of the media is managed by the disk drive.

  To minimize overheads, the disk drive performs a seek when required as part of the read or write order.

  The disk drive supports 'split' reads and writes when there will be a long delay before the specified starting block is reached. The drive recommends to the controller the LBA to be used in the next read or write order for the split. This can be used to minimize rotational latency by starting a transfer in the middle of the specified range of blocks.

  The disk drive reads the number of blocks specified in the order. Reading ahead data into the controller buffer can be managed by the controller issuing Extend Operation orders to extend the reading and transfer of data across the link. This read ahead can be quickly terminated by an Abort link control frame.
- The disk drive can receive an 'Extend Operation' order while executing a 'Write' order in order to execute back-to-back sequential writes without losing a revolution.
- The disk drive performs error recovery where possible for errors it detects. It can manage thresholds of errors recovered and automatically reassign blocks when thresholds for a block have been exceeded.

  Error information is returned in status frames to the controller.
- The Format order causes the disk drive to format the entire customer data area. The defect list created at manufacturing time and optionally the list of defects that have caused reassignments of blocks since that time are used to identify defective areas of the disk. These lists are held and managed within the disk drive.

  The block size to be used is defined within the Format order.
- The Reassign Block order causes the disk drive to reassign a logical block specified by the controller to a good area on the disk. The reassignment method is not specified and is determined by the disk drive.
- Orders are provided for the disk drive to report its device characteristics and vital product data.
- The disk drive controls the recording data channel, the actuator servo system and the spindle motor.

Conventions

Bits in a byte are numbered 7 to 0 from left to right, bit 7 being the most significant bit.

Hexadecimal values are represented by a trailing 'h', eg. 07h.

Future Extensions

This is the initial level of SSA-1 architecture. An extension to the architecture is being discussed that uses distributed frame switches to allow up to 16 nodes to be connected in either a daisy-chain or loop. Each node in this distributed switch configuration will require its own node identification. The definition of this function is not included in this level of the SSA-1 document, but provisions have been made in the frame definition to enable it to be added.

Link Management

The transport mechanism for transferring information across the serial link is defined in the SSA-0 document. Control frames and Application frames are described for communications across the link. This section defines the Control and Application frames and some control functions used for SSA-1.

Control frames

A control frame has a non-zero value in bits 7 to 2 of the control field. The Address field is ignored by the receiver except in a Link_reset frame when it is used to report the Link Status Byte. The frame must not include a data field. Control frames may be sent at any time since they are not subject to the normal pacing rules.

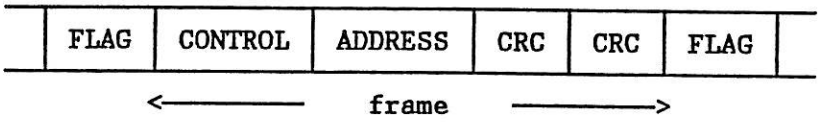
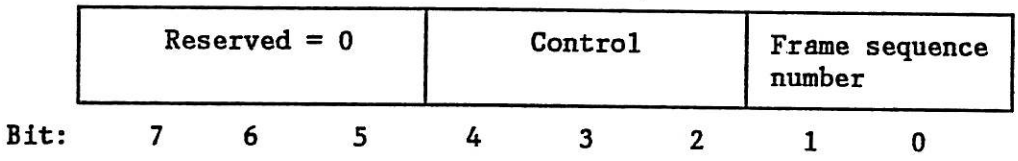


Figure 1. Format of a Control frame

In a Control frame, SSA-1 uses the control field as follows:



Reserved These 3 bits are not used in SSA-1 architecture.

Control These 3 bits are binary coded to indicate various control functions:

0 0 0 No control function. This is the normal value in Application frames.

0 0 1 This code indicates a Total_reset frame. The controller sends a Total_reset frame to force the disk drive to re-initialize its internal state. It is typically used to recover from a catastrophic error. See "Resets" on page 8 for details.

The disk drive does not issue Total_reset frames.

0 1 0 This code indicates a Link_reset frame. The Link_reset frame is issued by both the controller and the disk drive to report transmission errors and invoke the Link ERP. See the SSA-0 definition for details.

0 1 1 This code is reserved and is not used in SSA-1

1 0 0 This code indicates an Abort frame. The controller issues an Abort frame to terminate any order being executed by the disk drive. See "Resets" on page 8 for details.

The disk drive does not issue Abort frames.

1 0 1 This code indicates an Interrupt frame. The disk drive issues Interrupt frames repeatedly when it wishes to present status while disconnected.

The controller does not issue Interrupt frames.

1 1 0 This code is reserved and is not used in SSA-1.

1 1 1 This code is reserved and is not used in SSA-1.

Frame sequence Each node increments these 2 bits modulo 4 in each successive frame that it transmits. The Frame Sequence number is checked by the receiver in application frames and in an Abort control frame. It is used to initialize the controller's Receive Sequence Number in Interrupt frames. It is ignored by the receiver in the other control frames.

Application frames

An application frame has all zeros in bits 7 to 2 of the control field. SSA-1 uses 2 types of application frame - Message frames and Data frames.

Message frames

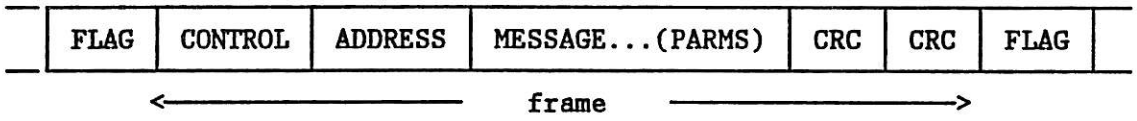


Figure 2. Format of a message frame

Message frames are used to communicate between the microprocessors in the controller and the disk drive. They are identified by the low order digit of the address field being hexadecimal 'F'. The first byte of the data field is a message code which identifies the function required. The length of the data field depends on the particular message. Each message is fully contained in a separate frame.

SSA-1 defines a controller-to-disk drive message for each order. It also defines disk drive-to-controller messages for requesting write data and returning status.

Data frames

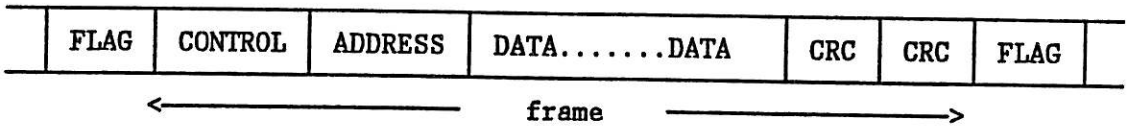


Figure 3. Format of a data frame

Data frames are used to transfer data between the controller and the disk drive. The destination node will have previously sent a Message frame to specify the value of the address field. The low order digit of the address field cannot be hexadecimal 'F' as this is used to identify message type frames. Typically this address will select a hardware DMA channel.

The length of the data field is variable. Transfer of a single logical block of data will typically require multiple data frames.

Protocol

The protocol for handling all SSA-1 orders except Extend Operation is as follows:

- The controller sends an order to the disk drive in a message frame.
- If the order requires the disk drive to transmit data then the first parameter of the order indicates the value to be placed in the address field of the data frames. When (and if) the disk drive wishes to transfer data it sends one or more data frames with the specified value in the address field.
- If the order requires the disk drive to receive data, a Ready_for_data message is returned when (and if) the disk drive wishes to receive data. The parameters of the message indicate the value to be placed in the address field of the data frames and the number of bytes that the disk drive is currently requesting. The controller then sends the requested number of bytes in one or more data frames with the specified value in the address field.

  This cycle of Ready_for_data messages and data frames may be repeated until the disk drive has obtained all the data that it requires.
- Finally the disk drive sends status in a message frame to the controller.

Except for Extend Operation, orders to the disk drive cannot be overlapped, that is, the controller may not send another order until status has been returned for the previous order. The protocol for the Extend Operation order is explained in "Extend Operation" on page 14. Formatting of the disk drive continues after good completion status has been returned to the Format order. Read Status orders can be sent to determine the progress of the formatting and when it has completed.

Resets

The control field in a link frame can request different types of reset to be performed. The functions for the following values of bits 4 - 2 of the control field are:

001 Total_reset.

This is used to reset the disk drive for a catastrophic error, such as a hardware error, link timeout or unrecoverable link error. The actions attempted by the drive include:

- Reset the hardware. If the disk is being written when the Total_reset is received, writing is stopped at a sector boundary in order to avoid creating a sector with bad ECC.
- Initialize the drive to be able to accept new orders. This may include a checksum test of the code RAM to ensure the code is valid, and a continuation of any initialization normally performed after a motor start. It may also include attempting to complete any reassignment that had not completed.
- Break reservation to any controller
- Set the disable link state, send DIS characters on the link and re-enable the link.

010 Link_reset

Link_reset frame is used to report transmission errors and invoke error recovery procedures (see SSA-0 for details).

100 Abort

This terminates any order being executed and clears any inbound data frames that may be buffered in the disk drive. If an order is terminated, an exception type status frame is returned to that order.

Spindle Synchronization

The link character K.28.0 is used for the Spindle Synchronization character in SSA-1. This character has the encoded value 0011110100 or its inverse 1100001011. A pair of these characters is sent from the controller at the nominal rotation rate of the spindle motor. The transmission of these characters can be interleaved with the transmission of frames on the link. Transmission may be delayed due to sending a pair of other protocol characters, such as RR and ACK, but is not delayed by the transmission of frames on the link. Synchronizing characters are used in pairs to protect from false decoding of other characters caused by transmission errors.

The drive keeps the spindle motor synchronized to these link synchronization characters with a defined skew after it has received a Set Position order. For further details of spindle synchronization see "Set Position" on page 16.

Orders

Orders are sent to the disk drive in message frames. The identification code of the order is defined in byte 0 of the link message.

The orders to a disk drive with their codes in parentheses are:

- Read (50)
- Write (51)
- Extend Operation (40)
- Motor Control (20)
- Set Position (21)
- Read Status (10)
- Code Download (42)
- Release (11)
- Format (41)
- Reassign Block (30)
- Read Device Characteristics (12)
- Read Vital Product Data (13)
- Diagnostic (22)

Codes for vendor unique orders should be in the range 80h to 9Fh.

Read

Byte 0	Code = 50h	
Byte 1	Return Address	
Byte 2 Byte 3 Byte 4 Byte 5	Logical Block Address	(msb) (lsb)
Byte 6 Byte 7	Count	(msb) (lsb)
Byte 8	Options	
Byte 9	Reserved	

The Read order invokes a seek operation to the required cylinder. The block identified in the logical block address (LBA) field is located. The Count field identifies the number of blocks to be read from the disk. The Return Address field contains the byte which is to be placed in the Address field of any data frames which are returned as a result of this order. Data is sent from the disk drive in data frames. If error recovery is required on the data read from the disk, this is performed in the drive before sending data on the link. A status frame is sent on completion of the order.

When the Options field specifies splitting is allowed (byte 8 bit 0 is zero), the 'tail' of the data may be requested to be read before the 'head'. This is achieved by the disk drive terminating the Read order with exception status 'condition not met' before any data is transferred. The disk drive can terminate with this exception status if it calculates from the location of the first block accessed when the head reaches the correct track that time can be saved if the 'tail' of the data is read before the 'head'. Bytes 3 - 6 of the exception status frame contain the LBA of the first block the controller should read in a subsequent Read order to read the tail of the data. The disk drive does not include any controller overhead in the calculation of the recommended LBA. The controller should issue another Read order to read the 'head' of the data.

A Count field of zero causes the actuator to seek to the required cylinder before returning a status frame without any data frames being transferred.

The options field bit definition is:

Bit Description

7 (msb) Report all errors

When this bit is one, details of the last error recovered in this order (including the count of the retries) are reported in the status frame returned at the end of the order.

When this bit is zero, errors recovered within this disk drive are not reported unless an error recovery threshold has been exceeded.

6 Limit recovery

When this bit is one, only limited recovery actions are attempted in the disk drive. This can be used to verify that data can be reliably read. If a data block cannot be recovered within this limited level of recovery, it may be possible to successfully read the block with the full recovery allowed when this bit is a zero. However, in this case, the block should either be re-written or reassigned.

When this bit is zero, all the error recovery steps supported by the drive may be attempted unless disabled by the Read Continuous bit.

5 Read Continuous

When this bit is one, data is transferred without adding delays for error recovery that would ensure data integrity. Erroneous data may be transferred.

When this bit is zero, all the error recovery steps supported by the drive may be attempted unless disabled by the Limit Recovery bit.

4-1 Reserved

0 Disable Split

When this bit is one, reading starts from the block identified in the order.The disk drive does not attempt to split the read operation.

When this bit is zero, the read order can be terminated with exception status 'condition not met' to allow the controller to read the 'tail' of the data and then the 'head'.

Write

Byte 0	Code = 51h
Byte 1	Reserved
Byte 2 Byte 3 Byte 4 Byte 5	(msb) Logical Block Address (lsb)
Byte 6 Byte 7	Count (msb) (lsb)
Byte 8	Options
Byte 9	Reserved

The Write order invokes a seek operation to the required cylinder. The block identified in the logical block address field is located. The Count field identifies the number of blocks to be written on the disk. The controller sends the appropriate number of data frames to the disk drive. The disk drive does not start writing each block until the whole block has been received in data frames on the link. A status frame is sent to the controller on completion of the order.

Data frame transfers are preceded by a Ready_for_Data' message from the disk drive. This message identifies the number of bytes that can be transferred. This should be no more than the buffer space available in the disk drive. If it has limited buffering, multiple Ready_for_Data messages are sent during the execution of the order, each message requesting data as space becomes available in the buffer in the disk drive.

When the Options field specifies splitting is allowed (byte 8 bit 0 is zero), data written to the media need not start at the block identified in the logical block address field of the order. This allows a split write operation by writing the 'tail' of the data blocks before writing the 'head' of the data. Data is requested starting from the block identified in the LBA field of the order by the Ready_for_Data message from the drive. If writing on the disk does not start at the block specified in the order, some sectors must be buffered in the drive.

Additionally, if splitting the writing to the disk is allowed by the Options field, the 'tail' of the data may be requested to be transferred before the 'head'. This is achieved by the disk drive terminating the write order with exception status 'condition not met' after the transfer of data requested by the Ready_for_Data message. The disk drive can terminate with this exception status if it calculates from the location of the first block accessed when the head reaches the correct track that time can be saved if the 'tail' of the data is written before the 'head'. Bytes 3 - 6 of the exception status frame contain the LBA of the first block the controller should send in a subsequent Write order to write the tail of the data. The disk drive does not include any controller overhead in the calculation of the recommended LBA. The controller should issue another Write order to write the 'head' of the data.

A Count field of zero causes the actuator to seek to the required cylinder before returning a good completion status frame. A Ready_for Data message is not sent and no data frames are transferred.

The options field bit definition is:

Bit	Description
7 (msb)	Report all errors
	When this bit is one, details of the last error recovered in this order (including the count of the retries) are reported in the status frame returned at the end of the order.
	When this bit is zero, errors recovered within this disk drive are not reported.
6-1	Reserved
0	Disable split
	When this bit is one, writing starts from the block requested in the Write Data order. The disk drive does not split the write operation or terminate the order with exception status to request the controller to split the write operation.
	When this bit is zero, the first block written need not be the block specified in the LBA field of the order. The disk drive is allowed to write the 'tail' of the data before the 'head'. The write order can also be terminated with exception status 'condition not met' to allow the controller to send separate write orders for the 'tail' and then the 'head' of the data.

Extend Operation

Byte 0	Code = 40h	
Byte 1	Reserved	
Byte 2 Byte 3 Byte 4 Byte 5	Logical Block Address	(msb) (lsb)
Byte 6 Byte 7	Count	(msb) (lsb)

This order extends the operation of a previous Read or Write order. The count field in the Extend Operation order indicates the number of additional blocks required to be read or written after the current order completes.

The Logical Block Address (LBA) field defines the address of the first block to be read or written. This will be one more than the LBA of the last block to be read or written by the previous order if contiguous reading or writing is required. If the LBA field is not the first block after the last LBA of the previous order, the blocks in between these LBA definitions are skipped over and not read or written. The LBA defined in the Extend Operation order must be higher than the highest block required in the previous read or write order or one that has been previously extended.

This order can be issued while the disk drive is still executing a previous read or write order without causing an error or exception status termination of the previous order. It can be used to execute without losing a revolution back to back write ocks operations for contiguous blocks in separate commands issued by the using system. It can also be used to read ahead data into the controller's buffer for data following that requested by a read operation.

If the Extend Operation order is received in time, a Read or Write type of order can receive multiple Extend Operations with a single completion status reported. If these extensions are for contiguous data from the previous Read, Write or Extend Operation order, there is no restriction on the time when each Extend Operation is issued. If the extension is for non-contiguous data, another Extend Operation cannot be issued until data transfer has started for the previous Extend.

The disk drive returns a status frame for the order that is being extended when all the extensions have completed. If the Extend Operation order is received by the disk drive too late to extend the order, a status frame is returned to the order that was in process. The controller determines from the status returned whether or not the Extend Operation was received in time and had been executed.

If an Extend order is received that overlaps any order other than a Read, Write, or one of these being extended, a status code of Invalid Order is returned. If an Extend Operation order is received after status has been returned for the last order, no status is returned to the Extend Operation order.

Motor Control

Byte 0	Code = 20h
Byte 1	Reserved
Byte 2	Options
Byte 3	Reserved

This order causes the spindle motor to be either started or stopped depending on the contents of the options field (byte 2).

Byte 2 bit 7 (msb) set to one requests the disk drive be made ready for use by spinning up the motor. This should be followed by any checkout of hardware only possible when the motor is running and the initialization of the drive, which may include the completion of a reassignment interrupted when the motor was stopped.

Byte 2 bit 7 (msb) set to zero requests the stopping of the spindle motor.

A good completion status frame is not returned until the motor has stopped or it has been spun up to full speed and all subsequent functions have completed.

Set Position

Byte 0	Code = 21h
Byte 1	Reserved
Byte 2	Position
Byte 3	Options

This order controls the synchronization of the motor to the spindle synchronization link characters from the controller. When bit 7 (msb) of the options field is zero, synchronization is enabled. Synchronization is disabled if bit 7 of the options field is one.

The rotational skew defined in the position field is applied in the retarded direction (lagging the spindle synchronization link characters from the controller. The value in the position field is the numerator of a fractional multiplier that has 256 as its denominator (e.g. a value of 64 indicates that the retarded skew delay of the drive from the spindle synchronization link characters is one quarter of a revolution). A value of zero indicates that there is to be no rotational offset from the spindle synchronization link characters.

When the disk drive has synchronized its motor relative to the spindle synchronization pulses as defined by this order, the synchronized bit is set in the status frame returned to a Read Status order. Synchronizing the spindle motor to the spindle synchronization characters on the link may not have been achieved before the status frame indicating good completion is returned to this order.

The motor is not synchronized following a power on or motor start. The disk drive does not attempt to synchronize to the spindle synchronization characters on the link until a Set Position order has been received. Once synchronized, total resets received on a link have no effect on the synchronization.

If the drive attaches to multiple links, it synchronizes to the characters sent on the link that last sent a Set Position order.

Read Status

Byte 0	Code = 10h
Byte 1	Reserved

This order can be used to receive the status frame from the drive without requesting any function to be executed by the drive. The drive returns a status frame for this order as defined in "Status Message" on page 25.

Byte 7 of the status frame contains generic status information:

Bit Description

7 (msb) Format in Progress

This indicates that a Format order has not yet completed. A progress indication is reported in byte 8 and 9. This 2 byte field (byte 8 is the most significant byte) is a fraction complete indication in which the returned value is the numerator that has 65536 as its denominator.

6 Synchronized

The spindle motor is synchronized to the synchronization characters sent on the link.

5 - 0 Reserved

Code Download

Byte 0	Code = 41h
Byte 1	Reserved
Byte 2 Byte 3 Byte 4 Byte 5	(msb) Count (bytes) (lsb)
Byte 6	Options
Byte 7	Code ID

This order is used to download a new code load to the disk drive and cause execution to switch to the new code once installed. The new code load is sent to the disk drive in data frames. The drive sends a Ready_for _Data message to identify the number of bytes that can be transmitted before data frames are sent. A status frame is returned on completion of the order.

The option field in byte 6 has the following definition:

Bit 7 (msb) Save code

When bit 7 is one, the code load will be loaded and also saved within the disk drive in non-volatile memory or on the media. If bit 7 is zero, the code loaded by this order is not retained after a power off.

Bits 6-0 Reserved

The Code ID field can be used to identify which portion of the code is being downloaded when the entire code is too large to be held in the drive as a single data set.

Release

Byte 0	Code = 11h
Byte 1	Reserved

This order releases the disk drive from the controller issuing the order.

Note there is not a corresponding 'Reserve' order as any order, other than Release, accepted by a disk drive involves an implicit reservation to the controller that issued the order. It is the controller's responsibility to release a disk drive that has been implicitly reserved to it to allow another controller to be able to communicate to the drive.

Reservation to a controller is also terminated when any controller issues a Total_reset to the disk drive. This permits another controller to break a connection between a controller and a disk drive when the controller that had reserved the drive has a failure.

Reservation to a controller is lost when the drive powers off. The drive powers on in a neutral state, in which it is not reserved to any link and can accept an order from any controller.

See "Multi-Port Attachment" on page 29 for more details on multi-port attachment.

Format

Byte 0	Code = 41h
Byte 1	Reserved
Byte 2	Reserved
Byte 3 Byte 4 Byte 5	(msb) Block length (bytes) (lsb)
Byte 6	Options
Byte 7	Reserved

The Format order causes the entire customer data area of the disk drive to be physically formatted. The length of each logical block is specified in the order. Good status is returned immediately the order has been accepted and validated. The controller can inquire on the progress of the format operation by issuing a Read Status order.

Defective areas of the media are by-passed by referring to two internal defect lists. The first is the Primary ('P' list) that is created at time of manufacturing. The second ('G' list) is a grown defect list consisting of entries corresponding to the reassignments due to Reassign Block orders and also due to automatic reassignments. The 'P' list can never be altered. Contents of the 'G' list can optionally be erased at format time.

The options field has the following definition:

Bit 7 (msb) P list only

When this bit is set to one, the grown defect list is erased and the disk drive formatted with the primary defect list only. When set to zero, the grown defect list is retained and the disk drive is formatted with both the primary and grown defect lists.

Bits 6 - 0 Reserved

The Format order must complete successfully for the disk drive to be usable. If the order is interrupted by a Total_reset or power down, the disk drive enters a degraded mode of operation and writing is prohibited. To exit the degraded mode, another Format order has to be sent and be successfully completed.

Reassign Block

Byte 0	Code = 30h
Byte 1	Reserved
Byte 2 Byte 3 Byte 4 Byte 5	Logical Block Address (msb) (lsb)

This order instructs the disk drive to reassign the specified logical block to a reliable area on the medium. The reallocation mechanism is not defined and it could involve pushdown of blocks to a good spare or direct reassignment to a spare. The disk drive updates the Grown Defect list when the block has been reassigned. A status frame is returned on completion of the reassignment.

It is the responsibility of the controller or the system to write the correct data in the logical block re-assigned.

The disk drive may move blocks on the media other than the block specified in the order to retain sequential ordering of blocks after the reassignment. All the blocks that will be re-written during the Reassign Block order are first read to determine if they can be read within a certain level of error recovery. If excessive recovery is required on any block, the reassignment process will not start and the order is terminated. The system should reassign the block identified in the status before attempting to reissue the Reassign Block order for the original block.

The reassignment process is checkpointed in the disk drive. If it cannot complete due to a power off or Total_reset, it attempts to continue with the reassignment on the next power on, Total_reset or Diagnostic order. The disk drive is not able to accept certain orders until the reassignment has been completed (e.g. Writes). These orders are terminated with Not Ready status, the status code indicating the reason for the degraded mode.

In addition to reassignments requested by this order, the drive may automatically reassign a block if successful recovery is required during a read or write order. This would be used when the level of recovery for that block is excessive and the integrity of data in the future is in doubt.

Read Device Characteristics

Byte 0	Code = 12h
Byte 1	Return Address

Device characteristics are returned in data frames to the link address specified.

The device characteristics data is 32 bytes which have the following definition:

Byte 0	Reserved	
Byte 1 Byte 2 Byte 3	Length of block (bytes)	(msb) (lsb)
Byte 4 Byte 5 Byte 6 Byte 7	Number of Blocks	(msb) (lsb)
Byte 8 Byte 9	Disk rotation time (microsecs)	(msb) (lsb)
Byte 10 to Byte 31	Reserved	

Read Vital Product Data

Byte 0	Code = 13h
Byte 1	Return Address

The vital product data is sent from the disk drive in data frames. The Return Address field contains the byte which is to be placed in the Address field of all data frames. The length of the data reported is 256 bytes.

The reserved fields are zeroes. All other fields are ASCII characters. If a field is not available, for example the field is contained on the media and the motor is stopped, then ASCII spaces are returned for those fields (except serial numbers are reported as ASCII zeroes).

The format of the data returned is shown in Figure 4.

Byte	Content
0	Additional length (bytes)
1 - 8	Vendor ID
9 - 15	Product Identification
16 - 24	Serial Number
25 - 28	Code Revision Level
29 - 255	Additional Product Data

Figure 4. Format of Vital Product Data

The additional length field is the binary count of the number of valid bytes that follow this field. Zeroes are reported in all bytes following the last valid byte.

The product identification field identifies the type of disk drive.

The serial number field uniquely identifies the attached drive. This serial number should change if the assembly that contains the media of the drive is changed.

Additional product data fields contain more information on the disk drive. This can include details on the type and level of the field replaceable units and information on the manufacturing time and place.

Diagnostic

Byte 0	Code = 22h
Byte 1	Reserved
Byte 2	Options
Byte 3	Reserved

This order causes internal diagnostics tests to be executed in the disk drive. A status frame is sent when the tests have completed. Failures detected during the tests are reported in the status frame.

When the options field is zero, a self test of the disk drive is requested. No other options are defined at present.

Messages from the Drive

Messages are sent from the disk drive to report status on completion of an order and to request the sending of data to the drive.

Status Message

Status is returned at the end of an order in a status type message frame.

Byte 0	Code = 01h
Byte 1	Reserved
Byte 2	Status Type / Code
Byte 3 Byte 4 Byte 5 Byte 6	(msb) Logical Block Address (lsb)
Byte 7 Byte 8 Byte 9 Byte 10 Byte 11 Byte 12	as defined for status type
Byte 13	Retry Count
Byte 14 Byte 15	Unit Error Code

The parameter bytes of the status frame consist of 16 bytes as follows:

Byte **Description**

0 Code

The code identifies this is a status frame.

1 Reserved

2 Status Type / Code

Bits	Description
7-5	Status type
4-0 (lsb)	Status code

3-6 Logical Block Address (LBA)

If the LBA field is not applicable or cannot be determined, 'FF'h is returned in bytes 3-6.

7-12 Additional information depending on the status type

13 Retry count

This is the count of the retry steps for the last recovered error in the order.

14-15 Unit error code

This supplies details of the error.

The following table defines the status type, their corresponding codes and any additional information in bytes 7 to 12 of the status frame.

Status Type	Status Code	
000	00000	Good completion of order. LBA = last block of Read/Write orders.
001		Reserved
010		Not Ready
	00001	Order not executable - Motor stopped
	00010	Order not executable - drive is in degraded mode
011		Unrecovered error
	00000	Seek error
		Medium error (status code = 01xxx) Bytes 7-10 contain the physical address of the last block read
	01001	ID field check or not found (LBA = block in error)
	01010	Data field ECC check (LBA = block in error)
	01011	Data field no sync found (LBA = block in error)
	10000	Hardware error
100		Recovered error
	00000	Seek error
		Medium error (status code = 01xxx) Bytes 7-10 contain the physical address of the last block read
	01001	ID field check or not found (LBA = block in error)
	01010	Data field ECC check (LBA = block in error)
	01011	Data field no sync found (LBA = block in error)
	11000	Error recovery threshold exceeded
	10000	Hardware error
101		Illegal Order
	00001	Overlapped Order
	00010	Invalid order
	00011	Invalid parameter

Status Type	Status Code	
110		Exception
	00001	Condition not met Bytes 3-6 contain the recommended LBA for next Read/Write
	00010	Order aborted
	00011	Reservation conflict
	00100	No spares available for reassignment
111		Reserved

Table 1. Status Frame Definition.

Ready_for_Data Message

Byte 0	Code = 02h
Byte 1	Return Address
Byte 2 Byte 3 Byte 4 Byte 5	Count (msb) (lsb)

This message is sent by the disk drive to request the amount of data defined in the count field. The disk drive should not request more data than it has buffer space available. This is to avoid the controller trying to send more frames than can be buffered in the drive which would result in the drive not sending Receiver Ready characters and so preventing the link being used for any other frames from the controller.

The return address field identifies the value to be placed in the address field of the data packets.

The count field identifies the number of blocks to be sent if the count field of the order in process identified blocks (eg. write order). It identifies the number of bytes to be sent if the count field in the order in process defined bytes (eg. Code Download).

Multi-Port Attachment

The disk drive can attach to multiple links to allow the drive to be used when one of the systems or attachments has failed. One of the controllers is considered to be a primary and the others are back-up. If the primary controller fails, then the using system can continue running by directing future accesses via a backup controller. It is not intended that the controllers share a disk drive dynamically, due to the difficulty of maintaining coherency between the buffers in the controllers.

The drive can be in one of the following states:

1. Neutral

   The neutral state is entered following a power on or Total_reset. In this state the drive is not reserved to any of the links and can accept an order on any link.

2. Reserved to a link

   A disk drive becomes implicitly reserved to a controller link when any order (other than Release) on that link is accepted for execution. There is no explicit reserve order.

   Reservation to a controller continues until one of the following events:

   a. A Release order is received from the same controller

   b. A Total_reset link control frame is received from any controller

   c. The disk drive is powered off

   When the disk drive is reserved to a controller, any order received from another controller is rejected with Exception status and a Reservation Conflict status code.

   When the system needs to switch control to the backup controller, reservation may have to be forced as the disk drive may still be reserved to the failed controller. This is achieved by the new primary controller sending a Total_reset link control frame, which releases all reservations, and then sending any order other than Release to reserve the drive implicitly.

   It is recommended that the primary controller issues a Release order periodically to allow the backup controller to send orders to the disk drive for configuration after it has powered on.