

Date: September 05, 2003

To: T10 Committee (SCSI)

From: Jim Coomes (Seagate)

Subject: SBC 32 Byte Commands for SBC 32 Byte Commands for End-to-End Data Protection

1 Overview

There is a need to provide the initial value of the DATA BLOCK REFERENCE TAG proposed in 03-176r4 on a command by command basis. One use case is in a configuration where a controller (e.g., a RAID) remaps the LBA to a different LBA space on a physical LUN. By passing the initial value of the DATA BLOCK REFERENCE TAG in the command to the LUN, the original data protection block may be passed through the controller to the LUN and checked. This function provides end to end protection in the remapping case.

To provide the space for the initial DATA BLOCK REFERENCE TAG and maintain 8 byte LBA space, 32 byte formats are proposed for read and write operations.

This proposal additionally provides a mechanism to enable device server checking of the DATA BLOCK APPLICATION TAG in the protection information.

Changes to document 03-176r4

Page 3,

The DATA BLOCK REFERENCE TAG field is **an incrementing value set to the least significant four bytes of the logical block address** to which the data block is associated. **For commands that do not include an initial DATA BLOCK REFERENCE TAG value**, the first data block transmitted shall contain the least significant four bytes of the logical block address contained in the LOGICAL BLOCK ADDRESS field of the command associated with the data being transferred. **For commands that include an initial DATA BLOCK REFERENCE TAG value, the first data block transmitted shall contain the 4 byte DATA BLOCK REFERENCE TAG value from the command.** Each subsequent data block's DATA BLOCK REFERENCE TAG field shall contain the data block reference tag of the previous data block plus one. of the logical block address contained in the LOGICAL BLOCK ADDRESS field of the command associated with the data being transferred. Each subsequent data block's LOGICAL ADDRESS TAG field shall contain the logical address tag of the previous data block plus one.

SBC-2 additions

1.0.1 READ (32) Command

The READ (32) command (see table 1) requests that the device server transfer data to the application client. The most recent data value written in the addressed logical block shall be returned

Table 1 — READ (32) command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (7Fh)							
1	CONTROL							
2	Reserved							
6								
7	ADDITIONAL CDB LENGTH (18h)							
8	(MSB)	SERVICE ACTION (TBDh)						(LSB)
9								
10	Reserved	RDPROTECT	DPO	FUA	Reserved	RELADR		
11	Reserved							
12	(MSB)	LOGICAL BLOCK ADDRESS						(LSB)
19								
20	(MSB)	INITIAL DATA BLOCK REFERENCE TAG						(LSB)
23								
24	(MSB)	DATA BLOCK APPLICATION TAG						(LSB)
25								
26	(MSB)	DATA BLOCK APPLICATION TAG MASK						(LSB)
27								
28	(MSB)	TRANSFER LENGTH						(LSB)
31								

The RDPROTECT field is defined in table 2.

Table 2 — RDPROTECT field

Value	Description
00b	Reserved
01b	Transmit the protection information. The device server may determine if the data block is valid by checking any of the protection information. If the device server determines there is an error as a result of checking the protection information the command shall be terminated with a CHECK CONDITION status and the sense data shall be set to MEDIUM ERROR with the additional sense code set to UNRECOVERED PROTECTION INFORMATION READ ERROR . A read command to a logical unit that supports protect information and has not been formatted with protection information may fail with a CHECK CONDITION status. If so the sense data shall be set to ILLEGAL REQUEST with the additional sense code set to INVALID FIELD IN CDB. If the logical unit does not support protection information and does not check the RDPROTECT field then the contents of the protection information is not defined by this standard.
10b	Transmit the protection information. The device server shall not check the contents of the DATA BLOCK GUARD field within the protection information. The device server may determine if the data block is valid by checking the DATA BLOCK APPLICATION TAG within the protection information. If the device server determines there is an error as the result of checking the protection information, the command shall be terminated with a CHECK CONDITION status and the sense data shall be set to MEDIUM ERROR with the additional sense code set to UNRECOVERED PROTECTION INFORMATION READ ERROR . A read command to a logical unit that supports protect information and has not been formatted with protection information may fail with a CHECK CONDITION status. If so the sense data shall be set to ILLEGAL REQUEST with the additional sense code set to INVALID FIELD IN CDB. If the logical unit does not support protection information and does not check the RDPROTECT field then the contents of the protection information is not defined by this standard.
11b	Transmit the protection information. The device server shall not check the contents of any of the protection fields. A read command to a logical unit that supports protection information and has not been formatted with protection information may fail with a CHECK CONDITION status. If a check condition occurs the sense data shall be set to ILLEGAL REQUEST with the additional sense code set to INVALID FIELD IN CDB. If the logical unit does not support protection information and does not check the RDPROTECT field then the contents of the protection information is not defined by this standard.

The INITIAL DATA BLOCK REFERENCE TAG field contains the value of the DATA BLOCK REFERENCE TAG expected on the first data block requested by the command. The device server may compare the DATA BLOCK REFERENCE TAG read from the medium with this value as enabled by the RDPROTECT field. in this command.

The DATA BLOCK APPLICATION TAG field contains a value that is expected in the protection information transferred by this command. The device server may compare the DATA BLOCK APPLICATION TAG read from the medium with this value as enabled by the DATA BLOCK APPLICATION TAG MASK.

The DATA BLOCK APPLICATION MASK field contains a value that is a bit mask for enabling the checking of the DATA BLOCK APPLICATION TAG in the protection information for each data block transferred by this command. A one in a DATA BLOCK APPLICATION TAG MASK bit enables the checking of the corresponding bit in the DATA BLOCK APPLICATION TAG field with the DATA BLOCK APPLICATION TAG read from the medium.

1.0.2 WRITE (32) command

The WRITE (32) command ((see table 3)) requests that the device server write the data transferred from the application client to the returned

Table 3 — WRITE (32) command

Byte\Bit	7	6	5	4	3	2	1	0
0	OPERATION CODE (7Fh)							
1	CONTROL							
2	Reserved							
6								
7	ADDITIONAL CDB LENGTH (18h)							
8	(MSB)	SERVICE ACTION (TBDh)						(LSB)
9								
10	Reserved	WRPROTECT	DPO	FUA	Reserved	RELADR		
11	Reserved							
12	(MSB)	LOGICAL BLOCK ADDRESS						(LSB)
19								
20	(MSB)	INITIAL DATA BLOCK REFERENCE TAG						(LSB)
23								
24	(MSB)	DATA BLOCK APPLICATION TAG						(LSB)
25								
26	(MSB)	DATA BLOCK APPLICATION TAG MASK						(LSB)
27								
28	(MSB)	TRANSFER LENGTH						(LSB)
31								

The WRPROTECT field is defined in table 4.

Table 4 — WRPROTECT field

Value	Description
00b	Reserved
01b	Preserve the contents of the protection fields (e.g., write to media, store in non-volatile memory). The device server shall determine if the data block is valid by checking the contents of the protection fields. If the device server determines there is an error as a result of checking the protection information the command shall be terminated with a CHECK CONDITION status and the sense data shall be set to MEDIUM ERROR with the additional sense code set to PROTECTION INFORMATION WRITE ERROR. If the logical unit supports protection information and has not been formatted to accept protection information the device server shall terminate the command with a CHECK CONDITION status and the sense data shall be set to MEDIUM ERROR with the additional sense code set to CANNOT WRITE MEDIUM - UNKNOWN FORMAT.
10b	Preserve the contents of the protection fields (e.g., write to media, store in non-volatile memory). The device server shall not determine if the data block is valid by checking the contents of the DATA BLOCK GUARD field of the protection information. The device server shall determine if the data block is valid by checking the DATA BLOCK APPLICATION TAG field within the protection information. If the device server determines there is an error as a result of checking the protection information the command shall be terminated with a CHECK CONDITION status and the sense data shall be set to MEDIUM ERROR with the additional sense code set to PROTECTION INFORMATION WRITE ERROR. In the case where the logical unit does not support protection fields and the device server does not check the WRPROTECT field the device servers response to the write command is not defined by this standard.
11b	Preserve the contents of the protection fields (e.g., write to media, store in non-volatile memory). The device server shall not check the contents of the protection fields. If the logical unit supports protection information and has not been formatted to accept protection information the device server shall terminate the command with a CHECK CONDITION status and the sense data shall be set to MEDIUM ERROR with the additional sense code set to CANNOT WRITE MEDIUM - UNKNOWN FORMAT.

The INITIAL DATA BLOCK REFERENCE TAG field contains the value of the DATA BLOCK REFERENCE TAG expected on the first data block trans by the command. The device server may compare the DATA BLOCK REFERENCE TAG received from the application client with this value as enabled by the WRPROTECT field. in this command.

The DATA BLOCK APPLICATION TAG field contains a value that is expected in the protection information transferred by this command. The device server may compare the DATA BLOCK APPLICATION TAG received from the application client with this value as enabled by the DATA BLOCK APPLICATION TAG MASK.

The DATA BLOCK APPLICATION MASK field contains a value that is a bit mask for enabling the checking of the DATA BLOCK APPLICATION TAG in the protection information for each data block transferred by this command. A one in a DATA BLOCK APPLICATION TAG MASK bit enables the checking of the corresponding bit in the DATA BLOCK APPLICATION TAG field with the DATA BLOCK APPLICATION TAG received from the application client.