To: T10 Technical Committee
From: Rob Elliott, Compaq Computer Corporation (Robert.Elliott@compaq.com)
Date: 28 September 2000
Subject: Bidirectional XDWRITEREAD commands for SBC-2

Revision 0: initial version

Revision 1: Requests 10-byte opcode 53h, fixes two typos (XPWRITEREAD). Updated section numbers to reflect sbc2r01. Removed the addition of XDWRITEREAD to the list of commands affected by the Maximum XOR data size field – since the target controls when to return read data, it is not necessary to limit the initiator's write size.

T10/00-309r1 by George Penokie, accepted into SAM-2 revision 14, modified SAM to allow protocols and commands to support bidirectional data transfers – commands which transfer both read and write data. This proposal adds 10-byte and 32 byte versions of an XDWRITEREAD command to SBC-2 that uses that feature.

The 10-byte opcode should be 53h. The 32-byte service action is up to the SBC-2 editor.

Changes to SBC-2 revision 1

Section 4.1 Table 1 – Service action code assignments

Add a service action code assignment row for XDWRITEREAD(32).

Section 4.2.1.9 Table 2 – (reservation table)

Add XDWRITEREAD (10)/(32) with the more restrictive of XDWRITE and XDREAD's rules for each reservation type – conflict, conflict, conflict, allowed, conflict, conflict.
[while there, change the formatting of XDWRITE (10/32) to XDWRITE (10)/(32).]

[new] 5.1.34 XDWRITEREAD (10) command

The XDWRITEREAD (10) command (see Table x) requests that the target XOR the data transferred (data-out) with the data on the medium and return the resulting XOR data (data-in). This is the equivalent to an XDWRITE (10) followed by an XDREAD (10) with the same Logical Block Address and Transfer Length. This command is only available on transport protocols supporting bidirectional commands.

Table x XDWRITEREAD (10) command

	7	6	5	4	3	2	1	0
0	Operation Code (53h)							
1	Rsvd	Rsvd	Rsvd	DPO	FUA	Disable Write	Rsvd	Rsvd
2	Logical Block Address							
3								
4								
5								
6	Reserved							
7	Transfer Length							
8								
9	Control							

See 4.2.1.9 for reservation requirements for this command.

See the XDWRITE (10) command (5.1.nn) and XDREAD (10) command (5.1.nn) for a description of the fields in this command.

[new] 5.1.34 XDWRITEREAD (32) command

The XDWRITEREAD (32) command (see Table x) requests that the target XOR the data transferred (data-out) with the data on the medium and return the resulting XOR data (data-in). This is the equivalent to an XDWRITE (32) followed by an XDREAD (32) with the same Logical Block Address and Transfer Length. This command is only available on transport protocols supporting bidirectional commands.

Table x XDWRITEREAD (32) command

	7	6	5	4	3	2	1	0
0	Operation Code (7Fh)							
1	Control							
2	Reserved							
3								
4								
5	Encryption Identification							
6	Reserved							
7	Additional CDB Length (18h)							
8	Service Action (NNNh)							
9								
10	Rsvd	Rsvd	Rsvd	DPO	FUA	Disable Write	Rsvd	Rsvd
11	Reserved							
12	Logical Block Address							
13								
14								
15								
16								
17								
18								
19								
20	Reserved							
21	Reserved							
22	Reserved							
23	Reserved							
24	Reserved							
25	Reserved							
26	Reserved							
27	Reserved							
28	Transfer Length							
29								
30								
31								

See the XDWRITEREAD (10) command (5.1.nn) and SPC-2 for a description of the fields in this command.

Section 4.2.3.2.1 Overview of storage array controller supervised XOR operations

Three XOR commands are needed to implement storage array controller supervised XOR operations: XDWRITE, XPWRITE, and XDREAD. *The XDWRITEREAD command may be used in place of a sequence of XDWRITE followed by XDREAD.*

Section 4.2.3.2.2 Update write operation

The update write operation writes user data to a device containing protected user data and updates the parity information on the device containing check data. The sequence is:

1) An XDWRITE command is sent to the device containing protected user data. This transfers the user write data to that device. The device reads the old user data, performs an XOR operation using the old user data and the received user data, retains the intermediate XOR result, and writes the received user data to the medium;
2) An XDREAD command is sent to the device containing protected user data. This command transfers the intermediate XOR data from the XOR device to the storage array controller;
3) An XPWRITE command is sent to the device containing check data. This transfers the intermediate XOR data (received in the previous XDREAD command) to the device containing check data. The device reads the old XOR data, performs an XOR operation using the old XOR data and the intermediate XOR data, and writes the new XOR result to the medium.

In place of steps 1) and 2), a single XDWRITEREAD command may be sent to the device containing protected data.

Section 4.2.3.2.3 Regenerate operation

The regenerate operation is used to recreate a data block that has an error. This is done by reading the associated data block from each of the other devices within the redundancy group and performing an XOR operation with each of these data blocks. The last XOR result is the data that should have been present on the unreadable device. The number of steps is dependent on the number of devices in the redundancy group, but the sequence is as follows:

1) A READ command is sent to the first device. This transfers the data from the device to the storage array controller;
2) An XDWRITE command with the DISABLE WRITE bit set is sent to the next device. This transfers the data from the previous read operation to the device. The device reads its.data, performs an XOR operation on the received data and its data, and retains the intermediate XOR result;
3) An XDREAD command is sent to the same device as in step 2. This transfers the intermediate XOR data from the device to the storage array controller;
4) Steps 2 and 3 are repeated until all devices (except the failed device) in the redundancy group have been accessed. The intermediate XOR data returned by the last XDREAD command is the regenerated user data for the failed device.

In place of steps 2) and 3), a single XDWRITEREAD command may be sent to the device.

Section 4.2.3.2.4 Rebuild operation

The rebuild operation is similar to the regenerate operation, except that the last XOR result is written to the replacement device. This function is used when a failed device is replaced and the storage array controller is writing the rebuilt data to the replacement device. The sequence is as follows:
1) A READ command is sent to the first device. This transfers the data from the device to the storage array controller;
2) An XDWRITE command with the DISABLE WRITE bit equal one is sent to the next device. This transfers the data from the previous read operation to the device. The device reads its data, performs an XOR operation using the received data and its data, and retains the intermediate XOR result;
3) An XDREAD command is sent to the same device as in step 2. This transfers the intermediate XOR data from the device to the storage array controller;
4) Steps 2 and 3 are repeated until all devices (except the replacement device) in the redundancy group have been accessed. The intermediate XOR data returned by the last XDREAD command is the regenerated user data for the replacement device.

5) A WRITE command is sent to the replacement device. This transfers the regenerated user data from step 4 to the replacement device. The replacement device writes the regenerated user data to the medium.

In place of steps 2) and 3), a single XDWRITEREAD command may be sent to the device.

Section 4.2.3.5.2 Buffer full status handling

When the storage array controller sends an XDWRITE or REGENERATE command to a device, the device has an obligation to retain the resulting XOR data until the storage array controller issues a matching XDREAD command to retrieve the data. This locks up part or all (depending on the size of the device's buffer and the size of the XOR data block) of the device's buffer space.

When all of the device's buffer is allocated for XOR data, it may not be able to accept new media access commands other than valid XDREAD commands and it may not be able to begin execution of commands that are already in the task set.

When the device is not able to accept a new command because there is not enough space in the buffer, the device shall terminate that command with a CHECK CONDITION status and the sense key shall be set to ILLEGAL REQUEST with the BUFFER FULL additional sense code.

When a storage array controller receives this status, it may issue any matching XDREAD commands needed to satisfy any previous XDWRITE or REGENERATE commands. This results in buffer space being freed for other commands. If it is a multi-initiator system and the storage array controller has no XDREAD commands to send, the storage array controller may assume the buffer space has been allocated to another initiator. The storage array controller may retry the command in the same manner that a command ending with TASK SET FULL status would be retried including not retrying the command too frequently.

The storage array controller may use command linking to avoid a buffer full condition. For example, a storage array controller supervised update write operation would consist of an XDWRITE command linked to an XDREAD command.

The bidirectional XDWRITEREAD command avoids the buffer full condition. The storage array controller may issue multiple XDWRITEREAD commands, since the device controls when it accepts more write data and provides read data.

Section 4.2.3.5.3 Access to an inconsistent stripe

…

a) When an XDWRITE *or XDWRITEREAD* command has been issued and completed, the device containing protected data has been updated but the device containing check data has not.

Section 5.1 Table 8 Commands for direct-access block devices

Add XDWRITEREAD (10) and XDWRITEREAD (32) to the list of opcodes. Mark them as Optional.

Section 6.2.11 XOR control page

[no change – XDWRITEREAD doesn't need to limit the write size, because the target can just stop accepting write data and return read data whenever its buffer is full, then resume acccepting write data later]

The MAXIMUM XOR WRITE SIZE field specifies the maximum transfer length in blocks that the target accepts for a single XDWRITE EXTENDED, XDWRITE, or XPWRITE command.

Annex A XOR command examples

Revision 1 has no section labels in this annex, so only the names are shown. Although the figures could probably be modified to show an XDWRITEREAD as an alternative to the XDWRITE followed by XDREAD, only text changes are suggested here.

Update write operation

…

Three SCSI commands are used: XDWRITE, XDREAD, and XPWRITE. *XDWRITEREAD may be used in place of any sequence of XDWRITE followed by XDREAD.*

Regenerate operation

…

Three SCSI commands are used: READ, XDWRITE, and XDREAD. *XDWRITEREAD may be used in place of any sequence of XDWRITE followed by XDREAD.*

Rebuild operation

…

Four SCSI commands are used: READ, XDWRITE, XDREAD, and WRITE. *XDWRITEREAD may be used in place of any sequence of XDWRITE followed by XDREAD.*

Third-party XOR Figure A.4 Update write operation

Not directly related to this proposal: This figure shows XDWRITE(16) where it should show XDWRITE EXTENDED(16).