X3T9.2/91-143

One Technology Way
Westford, Massachusettes 01886
508-392-2738

Concurrent
Computer Corporation

August 27, 1991

Gerry Houlder
Seagate Technologies
8001 E. Bloomington Freeway
M/S MPS 043
Bloomington, MN 55420

Dear Gerry;

Per our phone conversations, I am sending this letter. I have reviewed revision 3 of the dual port SCSI specification and feel I must add my two cents!

We are providing a fault resilient computer for use in transaction processing applications. We will have two systems which are both connected to mirrored, dual-ported drives. There will be an ethernet link between the systems providing minimal communications between the two systems.

My major area of concern with the current specification deals with the failure of one of the systems. Field Service comes in, powers off the failed machine, and repairs it. When the machine powers up, the H/W does a SCSI bus reset. The current version of the specification indicates that every reset from the machine being repaired will cause a total drive reset. The 'running' system currently supporting the load will have to recover from each reset caused by the machine under repair. The S/W driver should be able to cope with this situation, but now not only is the application running on only one computer, that computer is constantly retrying disk transfers that failed due to extra resets.

Proposed Changes...

The dual port implementation should be as unobtrusive as possible on all commands not directly related to accessing a port. Normal reads and writes should not be affected as long as the port is reserved. Therefore the following changes are being proposed:

A SCSI bus reset from either port will cause a unit attention condition for all initiators on *that* port. All commands, reservations, and synchronous negotiation agreements for *that* port are cleared and the other port continues to function normally. If the alternate port, is disabled, has not been active since power-on reset and / or is not currently reserved, then the SCSI bus reset will clear all commands, reservations and synchronous negotiations for both ports and will force a hard reset condition to the target.

This change will require the drives to maintain separate unit attention information and sense pages for each port. A request sense command must return the last sense for that port. When a port is selected, the sense pages for that port must be brought into context.

New Commands Required:

The nature of the system failure cannot be determined before it happens with 100% certainty. When one systems fails, the other system will try to take over the application. The 'slave' system in the process of becoming 'master', will 'force' the drive to it's port and this will make the old 'master' system get 'drive not ready'. The old 'master' system that is currently in the process of going belly up may still respond to interrupts, and therefore issue drive resets to each 'drive not ready' error. The current version of the spec would cause the drive to oscillate between the two ports. Therefore, several new commands are required to be able to force the connection to one port or another safely. The following functionality is required, it may be implemented in several commands as shown, or one command with several flags performing subfunctions.

Command	Description
Priority Reserve Port	Immediately reserve the drive for this port . Terminate any and all activity on the alternate port. Send an attention to the alternate port to indicate the termination. This command *forces* a *normal* reserve. Therefore, if the other port is enabled, the other port may do a priority reserve and get the port back. *If one command is used with several flags, it would allow a disable Alternate Port and a Priority Reserve to occur simultaneously, thereby, blocking any windows for the alternate port to regain its reservation.*
Enable Alternate Port	This will enable the alternate port after it was disabled via a Disable Alternate Port. Power up default is both ports enabled.
Disable Alternate Port	Disable the alternate port. All resets, commands, messages from the Alternate port will be ignored.
Get Port Status	Return following status information unless this port is disabled: Alternate port reserved Alternate port busy This Port reserved Alternate Port Disabled

Proposed use of the commands...

Whenever the drive will be accessed by a machine it should be reserved first. A reserve command is sent and is either honored, or fails with command time out. If the reserve fails, it can be retried later, or more drastic action like a Priority Reserve can be issued. If the Reserve completes successfully, the system can then go ahead and do all of its read and writes as it normally would. The drive will remain reserved until the system is done with the drive, this may be 200 msec or 2 days depending on the application. When the system is done with the drive, it should be nice and release it allowing the other system to reserve it.

If access is required through the other port, and the reserving machine has not released it in a long time, a message must be sent to the machine currently reserving the drive telling it to release the drive. This message may be sent via ethernet, sneaker net or some other medium. If the reserving system is not able to release the drive for some reason, (like it went south for the winter) the requesting machine can always do a Priority Reserve.

Priority Reserve is not a nice command. It basically rips the other port out of the water. Everything stops on the other side! It can even stop in the middle of writing a sector. If part of the disk had been buffered in

the other hosts memory, then the data on the disk may not be consistent. (Oh NO!!!, its *fsck* time again!). The Priority Reserve should be used as a last ditch effort to access the drive.

When the ultimate time comes, one system will die and the other system will try to take over running the application. The second system does a Priority Reserve and a Disable Alternate Port. Since the alternate port is disabled, it doesn't matter what you do to the failed system, the working system can run its application on that drive. The fact that the a Priority Reserve and a Disabled Alternate Port can occur simultaneously, prevents a window in which the system that is dying a slow death could re-reserve and / or reset the drive. (This can happen because a system that has failed may still respond to interrupts and carry out the driver error recovery routine).

The Disable Alternate Port command is also dangerous. If Field Service forgets to enable the alternate port after everything is fixed, you still don't have your second system available. That port is totally disabled, and will ignore almost everything. The software that issues the Disable Alternate Port command should probably stay in the background and issue a message to the console every hour reminding the operator that the port is disabled.

The Dual Port Status command is invaluable in determining the state of the ports without having to run a series of tests. This tool may be used during testing to determine exactly where the drive is. This may prevent leaving a port disabled by allowing Field Service to query drive status before leaving the site.

SMD and IPI dual port support:

The reserve, release, and priority reserve are supported on both SMD and IPI.
IPI supports Enable and Disable Alternate port.
SMD does not allow an unselected / reserved port to reset the drive.
IPI allows a reset from any enabled port to totally reset the drive.
Neither SMD nor IPI support the reserve timer.
SMD has an inactivity time-out.
IPI and SMD have a status indication of sorts.

Forethought and planning in this stage of the design / specification cycle will provide an excellent implementation of Dual Port. We now have the power to keep it simple and provide enough flexibility for our most critical users and applications.

Sheldon Kolansky

Principal H/W Engineer

CC: Bob Ellison
Kris Swanson
[redacted], NCR Corporation; 3718 N. Rock Road; Wichita, Kansas 67226
Jim Meile; Seagate Technologies; Short Hills Plaza; 636 Morris Tnpk.; Short Hills, N.J. 07078