Date: January 5, 1996

To: X3T10

From: Jim McGrath (408-894-4504)
             Quantum

Subject: SIP and Command Queuing

The following can be taken as a comment on SIP or as input into SIP-2, whatever is the pleasure of the committee. SAM requires a variety of functions be available to the highest software levels when the IO subsystem performs command queuing. However, the division of functions between various portions of hardware and software is interconnect specific. Thus parallel SCSI can remain SAM compliant while allowing targets the option to implement only simple command queuing.

The reason for this is many fold. First, for most applications simple command queuing is perfectly adequate - indeed, today the vast majority of implementations only use simple queuing. Second, the other queuing functions do not come for free. Every product generation requires a reverification of basic functionality. The more complex the features offered, the more interactions that are possible, the more complex the testing. This results in either long product introduction/qualification delays and/or less reliable products. The history of multiple initiator support in SCSI should be instructive on this score.

Note that all the more complicated functions required by SAM can be easily implemented through a combination of host adaptor/target actions. For instance, ordered or head of queue commands from the higher level are simple delayed at the host until the target has no other simple commands queued. Aborting commands should always be done on a complete, all commands at the target, basis. The host(s) can always re-enter the commands after that point, avoiding complicated error handling interactions between target and host.

Note that the requested action here only affects parallel SCSI, and does not make it illegal for targets to implement all the functions defined in SIP today. It does allow them the option to implement (and test) a simpler subset. Given the obvious market need, we believe that the standard should not make illegal a reasonable implementation with widespread possible use. The standard should allow both, and let the market decide exactly what products are desired.