



adaptec, inc.

X3T9.2/87-51

Memo to: John Lohmeyer, ANSI X3T9.2 Chairman
NCR Corp.
3718 N. Rock Road
Wichita, KS 67226

Memo from: Robert N. Snively
Adaptec
580 Cottonwood
Milpitas, CA 95035

Date: March 6, 1987

Subject: Proposal for Asynchronous Event Notification

Dear Mr. Lohmeyer:

There appear to be a number of special information exchanges that target devices may need to initiate to initiators. One such case that has been discussed in SCSI meetings is the requirement to notify a host computer that an operator has pressed a control button on a peripheral device. The control button may need to invoke certain actions from the host to either perform the requested function or to prepare the peripheral device to perform the requested function.

A second such case is the notification of certain failure modes in the peripheral device. Some failure modes, like power or over-temperature failures, may cause data integrity exposures unless prompt preventive actions are taken by one or more of the attached hosts.

Previous proposals to address these notification requirements have assumed a master-slave relationship between the host and the attached peripheral device. A unique characteristic of the SCSI is its capability of supporting full peer to peer communication. The unique peer to peer capability is a very natural way to meet the asynchronous event notification requirement. The following proposal uses the peer to peer capability to provide the notification function in an architecturally clean manner that is simple to implement. Note that it is fully consistent with the SCSI standard described in X3.131-1986 as well as the common command set.

The following text is proposed as new section 6.4.4 to describe how the normal SCSI interaction is used to provide asynchronous event notification.

6.4.4 Asynchronous Event Notification Example

Asynchronous event notification is required by some SCSI devices to inform Processor devices of the occurrence of an asynchronous event. Those devices that are to be notified of an asynchronous event must be Processor type devices with at least one LUN of address 0. These devices are typically computing system hosts. Those devices that wish to perform notification may be any type of SCSI device, but they must be capable of operating as initiators.

At system initialization time, those SCSI devices that may wish to perform asynchronous notification arbitrate as initiators and attempt to select each SCSI device which may be attached to the bus to determine if it is present. After a successful selection, the selecting SCSI device sends its IDENTIFY message to indicate that LUN 0 is to be addressed. The command transmitted is the INQUIRY command. Each device responds with its appropriate device type. The device that may wish to notify others then records which of the responding devices identify themselves as Processor type devices. Those devices are expected to be able to receive data transmitted by a SEND command. The notifying device must then clear any UNIT ATTENTION condition present in the Processor device and verify that the Processor device is ready to receive information by executing a TEST UNIT READY and a REQUEST SENSE command.

Those devices which are Processor devices must establish an internal process capable of correctly managing a data packet sent by any attached device which may transmit data via a SEND command to LUN 0. If the data packet management process is not active or not supported by a particular Processor device, the appropriate status or error condition will be presented if a SEND command is received. The status may be BUSY if the Processor device is temporarily unable to receive the SEND command. The error may be CHECK CONDITION with INVALID REQUEST indicated if the message does not meet the requirements of the data packet management process. Other error conditions may also be appropriate.

Upon detecting the occurrence of an event which must be signaled to one or more of the Processor devices, the notifying device prepares an informative data packet and transmits it to the appropriate Processor devices using a SEND command to LUN 0. [IT MAY BE APPROPRIATE TO DEFINE THE FORMAT OF THE DATA PACKET AS A TWO, FOUR, OR EIGHT BYTE PACKET WITH A SPECIFIED FORMAT. IF SO, THE FORMAT

153

WOULD BE SPECIFIED IN SECTION 11:2 OF THE SPEC:] The Processor device examines the packet and, if the information it contains is valid, responds with GOOD status and a COMMAND COMPLETE message. The data packet management process then determines what commands should be sent to the notifying device to manage the indicated asynchronous event. The sequence of commands required to manage the event is unique to the particular type of device and is not specified by this document.

I am confident that the above example will explain and simplify the asynchronous event notification process. The requirement to support both target and initiator roles falls only on those devices which choose to participate in the protocol. All others are not even aware that any part of the protocol has taken place. The present set of SCSI protocol chips simply supports both target and initiator protocols. At the same time, the protocol allows Processor devices to signal asynchronous events to other Processor devices, a function probably far more important than any peripheral device notification. This allows host computers to pass responsibility for various activities back and forth to each other depending on the availability of resources or the occurrence of errors.

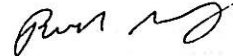
If an information packet format is required, the following format should not normally be confused with non-notification data packets. The format description will be placed in section 11.3 as follows:

The asynchronous event notification procedure described in section 6.4.4 uses the SEND command to transmit a standard notification packet. The standard notification packet has the following format. The format is selected to have a low probability of matching any normal packet format header. Such a packet will be transmitted whenever a device capable of performing a notification operation has determined that an asynchronous event has occurred. The packet is directed to the Processor devices that the notifying device has previously located through an INQUIRY command. The notification code is a 16 bit value that indicates in a vendor unique manner the occurrence of one or more asynchronous events.

Byte 0	FF h
Byte 1	FF h
Byte 2	FF h
Byte 3	FF h
Byte 4	Reserved
Byte 5	Reserved
Byte 6	(MSB) Notification Code
Byte 7	Notification Code (LSB)

NOTIFICATION INFORMATION PACKET (FIGURE 11-7)

Thank you for your acceptance of this proposal.



Robert N. Snively

154