Proposal for Parallel SCSI: Domain Validation

To: T10 Technical committee
From: Jim McGrath
Quantum Corporation
500 McCarthy Boulevard
Milpitas, CA USA 95035
Phone: 408-894-4504
Fax: 408-952-3620
Email: jim.mcgrath@quantum.com
Date: May 19, 1998

0 Background

This proposal is the outgrowth of discussions on the topic of expanders at the last T10 working group meeting. I have circulated drafts for comment to various people over the last few days, and I'd like to thank those who have taken the time to review the drafts and give me feedback. As usual, all errors are mine.

Since the key intent of this proposal is the creation of an architecture for domain validation testing, and to resolve the issue of detecting old expanders which do not understand double edge clocking, I have deliberately left out some more controversial items that we have discussed previously. If we can get agreement on the key items here, then we can discuss other possible enhancements to the standard later. Note that many of those items can be done with just this architecture in the standard (i.e. the architecture gives implementers a lot of freedom to innovate in this area and still maintain product interoperability).

After the meeting it was brought to my attention that the initiator should have the opportunity to conduct its own domain validation tests. This required the addition of a message to get the target's attention – which in turn could also be used by the target to clearly identify a domain validation. The sequence is still terminated with an affirming IUTR negotiation (or a new negotiation in the case of a failure).

This proposal is not structured as an explicit modification to SPI. Since there is still some disagreement on the overall content, I felt it was best to focus on that with this revision. If we can get enough agreement, the next revision will be structured as a formal modification to the SPI-3 draft document.

1 Introduction

This document describes a proposal for the SCSI parallel interface to:

1) Create an architecture to test elements of the SCSI domain (principally the cable plant) in order to determine if the data transfer parameters negotiated between the target and the initiator can be supported in their domain;
2) Specify one test for double-edged clocking and CRC;

The methods defined in this proposal allow initiators and targets to be backward compatible with SPI-2 compatible initiators and targets. This proposal may be implemented in hardware, firmware, or a combination of both hardware and firmware.

2 Definitions

2.1 CRC (Cyclic Redundancy Check): An error detecting code used to detect the validity of data that has been transferred during the current DEC DATA IN or DEC DATA OUT phase.

2.2 DEC (double-edged or dual-edged clocking): The method for transferring data into a register or latch on both polarity edges of the clock signal. Double-edged clocking is used in the DEC DATA phases to perform data transfers on both edges of the REQx or ACKx. Double-edged clocking shall occur only when the DEC DATA OUT and DEC DATA IN phases are in effect. The DEC DATA OUT phase is in effect when MSG is asserted, C/D is negated and I/O is negated. The DEC DATA IN phase is in effect when MSG is asserted, C/D is negated and I/O is asserted.

> Note: These two phase cases were reserved in SPI-2. By using previously reserved cases, low level hardware such as expanders and data capture tools can determine whether double edge clocking is being used and in what direction the DB(Px) line should be driven.

2.3 Initiator Domain Validation Memory (IDVM): The amounts of memory for the initiator, expressed in transfers on the SCSI bus (bytes for an 8-bit bus, words for a 16-bit bus, and double words for a 32-bit bus). [Note: this could be in bytes, but specified as transfers it made the Domain Validation section easier to write.]

2.4 Target Domain Validation Memory (TDVM): The amount of memory for the target, expressed in transfers on the SCSI bus (bytes for an 8-bit bus, words for a 16-bit bus, and double words for a 32-bit bus). [Note: this could be in bytes, but specified as transfers it made the Domain Validation section easier to write.]

3 Initiating Domain Validation Sequence

Another proposal documents the process by which new protocols and speeds are negotiated using the IUTR message. After these protocols and speeds have been negotiated, a sequence of one or more domain validations may be performed if the Domain Validation bit in the IUTR message was set.

In this case the target and initiator shall have also indicated through the IUTR negotiation the amount of high speed memory (i.e. capable of being filled or emptied at the negotiated transfer rate) available for domain validation.

Domain Validation may be performed for the following items:

- Protocol: to validate that double-edged clocking and CRC protocol negotiated between the target and the initiator is supported by the domain containing the target and the initiator.

4 Domain Validation

Either the initiator or the target may initiate a Domain Validation. A Domain Validation is initiated by the sending of a DOMAIN VALIDATION START message. This uniquely identifies the start of each Domain Validation. It is then followed by a DEC DATA IN phase, or a pair of DEC DATA IN and DEC DATA OUT phases. It is terminated by either another DOMAIN VALIDATION START message (to start another Domain Validation) or another IUTR negotiation.

Note that multiple Domain Validations may have to be executed in order to validate a domain. A Domain Validation may fail, and yet still result in the validation of a domain. The device that initiates the Domain Validation is responsible for making the decision as to whether the domain is valid or invalid.

4.1 Target Initiated Domain Validation

The following is the sequence for target initiated Domain Validation. Note that the DEC/CRC protocol validation is unique in that only steps A, B, and C are performed (see 4.3.1 for details).

a) The target initiates a domain validation by going into MESSAGE IN phase and transferring a DOMAIN VALIDATION START message;
b) the target enters a DEC DATA IN phase;
c) The target transfers the IDVM or TDVM number of data transfers, whichever is fewer, to the initiator (the data transferred is vendor unique and CRC information is not counted towards this total);
d) The target enters a DEC DATA OUT phase;
e) The initiator transfers the IDVM or TDVM number of data transfers, whichever is fewer, to the target (the data transferred is the data received by the initiator from the target in the previous DEC DATA IN phase and is transferred in the order received by the initiator; once again, the CRC information is not counted towards this total);
f) The target then compares the data received from the initiator with the data sent by the target during the previous DEC DATA IN phase;
g) Based on the outcome of this comparison, previous Domain Validations, and any exception conditions encountered, the domain may be validated or invalidated for the item being tested (see clause 4.3 for specific domain validations).

4.2 Initiator Initiated Domain Validation

The following is the sequence for initiator initiated Domain Validation.

a) The initiator initiates a domain validation by creating an ATTENTION CONDITION and transferring a DOMAIN VALIDATION START message in the subsequent MESSAGE OUT phase;
b) The target enters a DEC DATA OUT phase;
c) The initiator transfers the IDVM or TDVM number of data transfers to the target (the data transferred is vendor unique and CRC information is not counted towards this total);
d) The target enters the DEC DATA IN phase;
e) The target transfers the IDVM or TDVM number of data transfers, whichever is fewer, of data to the initiator (the data transferred is the data received by the target from the initiator in the previous DEC DATA OUT phase and is transferred in the order received by the target; once again, the CRC information is not counted towards this total);
f) The initiator compares the data received from the target with the data sent by the initiator during the previous DEC DATA OUT phase;

g) Based on the outcome of this comparison, previous Domain Validations, and any exception conditions encountered, the domain may be validated or invalidated for the item being tested (see clause 4.3 for specific domain validations).

4.3 Specific Domain Validations

4.3.1 Validating the Domain for DEC/CRC Protocol

The purpose of this validation is to detect elements of the domain that do not recognize the DEC/CRC protocol (e.g., old expanders).

The Domain Validation shall be performed as specified in clauses 4.1 or 4.2, except that only the DEC DATA IN portion is done and the number of words transmitted shall be reduced by a factor of two. Specifically:

a) The target initiates a domain validation by going into MESSAGE IN phase and transferring a DOMAIN VALIDATION START message;
b) the target enters a DEC DATA IN phase;
c) The target transfers IDVM or TDVM number of data transfers to the initiator (the data transferred is vendor unique and CRC data is not counted towards this total);
d) The initiator performs the normal CRC and parity (if a 16 or 32 bit transfer) check on the data received from the target;
e) Based on the outcome of this comparison and any exception conditions encountered, the domain is validated or invalidated for the DEC/CRC protocol.

This Domain Validation shall be initiated by at least one of the devices whenever the DEC/CRC protocol has been negotiated. It shall be done before any other Domain Validation.

5 Terminating Domain Validation Sequence

Any device which determines that the domain has been invalidated for an item agreed to during the previous IUTR negotiation shall consider the results of the previous IUTR negotiation to be invalidated at that point, and shall initiate another IUTR negotiation. In this case the device shall exclude from consideration during this negotiation the item for which the domain was invalidated.

The device that initiated the IUTR negotiation may terminate Domain Validation. This is done by initiating another IUTR negotiation. If both devices agree that all of the items requiring Domain Validation have been validated, then the Domain Validation bit shall be cleared to zero and the target shall go into whatever SCSI phase is appropriate for the command being executed after the IUTR negotiation. Otherwise the Domain Validation bit shall be set to 1 and another Domain Validation Sequence will begin.

The time to execute all of the IUTR negotiations and Domain Validation Sequences during a given I-T connection shall not exceed 100 ms.

6 Exception Conditions during Domain Validation

6.1 CRC/Partiy Error

A CRC or a parity error (for a 16 bit or 32 bit DATA BUS).

6.2 Data Pattern Mismatch

A mismatch between the data that the device that initiated the Domain Validation sent to the other device, and the data received from that other device.

6.3 Non-zero REQ/ACK Offset

If the target detects a stable non-zero REQ/ACK offset and no data transfer for at least 1 ms, then the device shall clear its REQ/ACK offset to 0 and shall go into MESSAGE IN phase. On the subsequent MESSAGE IN phase, the target shall send a REQ/ACK OFFSET MISMATCH DETECTED message.

If the initiator detects that the target has switched phases when the REQ/ACK offset is not zero, then it shall clear its REQ/ACK offset to 0 and shall create an ATTENTION CONDITION. On the subsequent MESSAGE OUT phase, the initiator shall send a PHASE CHANGE DETECTED WITH A NON-ZERO REQ/ACK OFFSET message.

6.4 Domain Validation Sequence Timeout

The time to execute the Domain Validation Sequence shall not exceeds 100 ms. If the time exceeds 100 ms, then the domain shall be considered invalidated and the Domain Validation Sequence shall be terminated.

6.5 Unexpected Protocol

If the target receives any message other than those specified in this section, then the domain shall be considered invalidated and the Domain Validation Sequence shall be terminated.

If the initiator detects any phase change other than those specified in this section, then the domain shall be considered invalidated and the Domain Validation Sequence shall be terminated.

If the target goes to the BUS FREE phase for any reason before the Domain Validation Sequence has been terminated, then the domain shall be considered invalid.