Document: T10/00-354r2 Date: 3 November 2000
To: T10 Committee Membership
From: Edward A. Gardner, Ophidian Designs
Subject: 64-bit Tags in SVP

When discussing SVP with the IBTA Application Working Group, several members requested that SVP commands and responses include a 64-bit unique identifier. The AWG requested that I propose this to T10.

The desire is that every SVP command IU contain a 64-bit identifier that would be returned in the corresponding response IU. Host driver software would use this to uniquely identify all outstanding IO operations across all Initiators (adapters), Targets and LUNs. SVP's current 32-bit Task Tags were intended to allow this, but members of AWG felt that field size would be too small for future large scale systems.

As described in revision 0 of this document, the simplest way to accomplish this is to increase SVP's TAG field size to 64 bits. That change was approved at the T10 CAP Working Group on September 13, 2000 and at the T10 Plenary meeting on September 14, 2000.

However, that working group meeting also indicated a strong desire to take advantage of the 64-bit TAG being unique among all requests from an Initiator, not just within an I_T_L nexus as (minimally) mandated by SAM. This allows the LUN field to be removed from the SVP_RSP information unit, and repositioned within the SVP_CMD information unit to better match FCP. Revision 1 of this document describes the changes to incorporate that recommendation into SVP.

Revision 2 of this document incorporates additional changes resulting from the T10 CAP Working Group on November 1, 2000.

I encountered two questions while drafting revision 1 that were not clear from the previous working group discussion. The first is whether targets check or enforce that the TAG field value is unique. Per the direction of the November 1 working group, SVP will specify that initiators **shall** supply unique TAG field values and that targets **are not required** to check whether the values are unique.

The other issue is the ABORT TASK request. Specifically, within an SVP_CMD information unit that contains an ABORT TASK request, which field contains the TAG of the task to be aborted? The seemingly obvious choice that the normal TAG field contains the TAG value to be aborted results in ABORT TASK being an exception. The resulting description of the TAG field would read something like:

> The TAG value of an SVP_CMD request containing an ABORT TASK task management function shall match the TAG value of the SVP_CMD request that contained the task to be aborted. An initiator is expected to provide a TAG value in other requests that is unique among all of the initiator's outstanding requests.

This exception would be reflected in the SVP driver code. When an ABORT TASK is issued, sometimes the initiator will first receive an ABORT TASK response, sometimes a response to the task that was to be aborted (if the target sent the task response before it received the ABORT TASK). The initiator must distinguish these two responses, as in the latter case the ABORT TASK response is still pending and the initiator cannot re-use the TAG value (and associated data structures) until after that response arrives.

Unfortuneately, as presently defined for SPI-n and FCP-n, the responses for a successfully completed task and a successfully completed task management request are identical. While it is necessary for the initiator to distinguish these two responses, at present there is no way to do so. Note that FCP-n does not have this problem because it uses an FC-2 layer abort exchange operation for ABORT TASK rather than an information unit. SPI-n does not have this problem because it is an interlocked bus.

We could solve this problem by inventing a unique status or RESPONSE DATA code to indicate successful completion of an ABORT TASK request. The approach I prefer and have described below is to assume that the TAG field is unique for all requests, without an exception for ABORT TASK. The TAG of the task to be aborted

would be conveyed in some other field of an ABORT TASK request. The first eight bytes of the CDB field seem the obvious choice.

The November 1 working group approved this approach, however Bob Snively requested that the CDB and task management request information unit formats be documented in different tables. After consideration I concluded that this was best accomplished by separating the current text into two sub-clauses, one for each information unit format.

The remainder of this document describes the proposed changes to SVPr01.

Table 1 replaces Table 2 in SVPr01, it summarizes the changes for all IUs. The order of the REQUESTLIMITDELTA and TAG fields is reversed, and TAG extended to eight bytes. Note that bytes 12-15 were formerly reserved in every IU, they are now part of the TAG field.

Table 1 - Fields common to all information units

Bit Byte	7	6	5	4	3	2	1	0
0				TYPE				
1								
2				RESERVED				
3								
4	MSB							
• • •				REQUESTLIMITDELTA (ONLY WHEN SENT BY TARGET)				
7								LSB
8								
• • •				TAG				
15								
16								
• • •				varies				
n								

Replace the last two paragraphs of clause 5.1 with the following:

Bytes 4 through 7 of each IU sent by an SVP target contain REQUESTLIMITDELTA. See clause 4.3 for a descripton of that field's use. Those bytes are used for other purposes in IUs sent by an SVP initiator.

Bytes 8 through 15 of each IU contain a TAG value, which provides a mechanism for matching requests with their corresponding responses. A requestor shall provide a TAG value in each request that is unique among all of the requestor's outstanding requests. A respondor shall copy the TAG value from each request to its response. Responders are not required to check whether the TAG values of outstanding requests are unique.

Table 2 replaces Table 6 in SVPr01, it shows the revised SVP_RSP IU format. In addition to the TAG and REQUESTLIMITDELTA field changes described above, the LOGICAL UNIT NUMBER field has been deleted.

The remainder of this document is a complete replacement for clause 5.2 in SVPr01. It has been split into two sub-clauses, one for information units containing CDBs and one for task management requests. All insertions or deletions from SVPr01 are marked with change bars. Blocks of text that have merely been moved are not marked with change bars. This incorporates the extension to the ADDITIONAL CDB LENGTH field that came out of the SPC-2 / FCP-2 comment resolution; that change was not present in previous versions of this document.

Table 2 - SVP_RSP information unit

Bit Byte	7	6	5	4	3	2	1	0
0	TYPE							
1				RESERVED				
2								
3								
4	MSB							
• • •				REQUESTLIMITDELTA				
7								LSB
8								
• • •				TAG				
15								
16				RESERVED				
17								
18	RESERVED				RIDUNDER	RIDOVER	SNSVALID	RSPVALID
19	STATUS							
20	MSB							
• • •				RESIDUAL COUNT				
23								LSB
24	MSB							
• • •				SENSE DATA LIST LENGTH = n				
27								LSB
28	MSB							
• • •				RESPONSE DATA LIST LENGTH = m				
31								LSB
32	MSB							
• • •				RESPONSE DATA (m bytes long)				
31+m								LSB
32+m	MSB							
• • •				SENSE DATA (n bytes long)				
31+m+n								LSB

5.2   SVP_CMD information unit

5.2.1   SVP_CMD information unit containing a CDB

An SVP_CMD information unit conveys a SCSI task (CDB) if the TASK MANAGEMENT FLAGS field contains zero (see table 3). SVP_CMD information units that contain a CDB whose length is sixteen bytes or less shall be sent as a 64 byte VI message. SVP_CMD information units that contain a CDB whose length is greater than sixteen bytes shall be sent as a VI message whose length is the CDB length plus 48 bytes. SVP initiators shall not issue commands with CDB sizes that would require sending an SVP_CMD information unit longer than INIMAXCMDIU.

Table 3 - SVP_CMD information unit containing a CDB

Bit Byte	7	6	5	4	3	2	1	0
0	TYPE							
1	RESERVED							
2								
7								
8	TAG							
•••								
15								
16	MSB							
•••	DATA VIRTUAL ADDRESS							
23								LSB
24	MSB							
•••	DATA MEMORY HANDLE							
27								LSB
28	MSB							
•••	DATA LENGTH							
31								LSB
32	RESERVED							
•••								
35								
36	MSB							
•••	LOGICAL UNIT NUMBER							
43								LSB
44	RESERVED							
45	RESERVED					TASK ATTRIBUTE		
46	TASK MANAGEMENT FLAGS							
47	ADDITIONAL CDB LENGTH = (N-63)/4						RDDATA	WRDATA
48	MSB							
•••	CDB							
63								LSB
64	MSB							
•••	ADDITIONAL CDB							
n								LSB

> NOTE 6 - SPC-2 requires that all CDBs longer than sixteen bytes be a multiple of four bytes in length, implying that all SVP_CMD information units will also be a multiple of four bytes in length.

The TYPE field shall contain the value specified in table 1.

The LOGICAL UNIT NUMBER field specifies the address of the logical unit of the I_T_L_Q nexus for the current task. The structure of the logical unit number field shall be as defined in the SCSI Architecture Model-2 standard. If the addressed logical unit does not exist, the task manager shall follow the SCSI rules for selection of invalid logical units as defined in the SCSI Primary Commands-2 standard.

The DATA VIRTUAL ADDRESS field contains the starting virtual address for the initiator data buffer that the target shall use as the source or destination of any RDMA operation requested for the command. The contents of DATA VIRTUAL ADDRESS shall be ignored if DATA LENGTH contains zero. The DATA VIRTUAL ADDRESS field shall be transferred from most significant byte first to the least significant byte last.

The DATA MEMORY HANDLE field contains the Memory Handle for the initiator data buffer that the target shall use as the source or destination of any RDMA operation requested for the command. The contents of DATA MEMORY HANDLE shall be ignored if DATA LENGTH contains zero. The DATA MEMORY HANDLE field shall be transferred from most significant byte first to the least significant byte last.

The DATA LENGTH field contains a count of the greatest number of data bytes expected to be transferred to or from the application client data buffer by the SCSI CDB. The parameter is the command byte count defined by ANSI X3.270. A DATA LENGTH value of 0 indicates that no data transfer is expected regardless of the state of the RDDATA and WRDATA bits and that no RDMA operation(s) shall be requested for the command. DATA LENGTH shall contain zero if the SVP_CMD IU contains a CDB that does not perform a data transfer. The DATA LENGTH field shall be transferred from most significant byte first to the least significant byte last.

> NOTE 7 - The byte transfer order of the DATA VIRTUAL ADDRESS, DATA MEMORY HANDLE and DATA LENGTH fields is the reverse of that described in Virtual Interface Architecture Specification, V1.0.

The TASK ATTRIBUTE field is defined in table 4.

TABLE 4 - TASK ATTRIBUTE

Codes	Description
000b	Requests that the task be managed according to the rules for a simple task attribute. (See SAM-2)
001b	Requests that the task be managed according to the rules for a head of queue task attribute. (See SAM-2)
010b	Requests that the task be managed according to the rules for an ordered attribute. (See SAM-2)
011b	Reserved
100b	Requests that the task be managed according to the rules for a automatic contingent allegiance task attribute. (See SAM-2)
101b-111b	Reserved

The TASK MANAGEMENT FLAGS field shall be zero for SVP_CMD information units that contain a CDB.

The ADDITIONAL CDB LENGTH field contains the length in 4-byte words of the ADDITIONAL CDB field.

RDDATA, when set to 1, specifies that the initiator expects the command to transfer data from the target to the initiator. This is a SCSI read-type operation.

WRDATA, when set to 1, specifies that the initiator expects the command to transfer data from the initiator to the target. This is a SCSI write-type operation.

If RDDATA and WRDATA are both set to 0, there shall be no data transfer and DATA LENGTH shall contain zero. The initiator shall not set RDDATA and WRDATA both to 1.

The CDB field contains the actual CDB to be interpreted by the addressed logical unit. Any bytes between the end of a 6 byte CDB, 10 byte CDB or 12 byte CDB and the end of the CDB field shall be reserved.

The ADDITIONAL CDB field contains any CDB bytes beyond those contained within the standard 16 byte CDB field.

The contents of the CDB and ADDITIONAL CDB fields shall be as defined in the SCSI command standards.

5.2.2   SVP_CMD information unit containing a task management request

An SVP_CMD information unit conveys a task management request if the TASK MANAGEMENT FLAGS field contains a non-zero value (see table 5). SVP_CMD information units that contain a task management request shall be sent as a 64 byte VI message.

Table 5 - SVP_CMD information unit containing a task management request

Bit Byte	7	6	5	4	3	2	1	0
0				TYPE				
1								
2				RESERVED				
7								
8								
•••				TAG				
15								
16								
•••				RESERVED				
35								
36	MSB							
•••				LOGICAL UNIT NUMBER				
43								LSB
44				RESERVED				
45				RESERVED				
46				TASK MANAGEMENT FLAGS				
47				RESERVED				
48								
•••				TAG OF TASK TO BE MANAGED				
55								
56								
•••				RESERVED				
63								

The TYPE field shall contain the value specified in table 1..

The LOGICAL UNIT NUMBER field specifies the address of the logical unit component of the nexus for the task management request. The structure of the logical unit number field shall be as defined in the SCSI Architecture Model-2 standard. This field is reserved if the task management request is not directed to either an I_T_L or I_T_L_Q nexus.

The TASK MANAGEMENT FLAGS field is defined in table 6.

TABLE 6 - TASK MANAGEMENT FLAGS

Codes	Description
00h	The SVP_CMD IU contains a command, not a task management request.
01h	The task manager shall perform an ABORT TASK function (see SAM-2).
02h	The task manager shall perform an ABORT TASK SET function (see SAM-2).
04h	The task manager shall perform a CLEAR TASK SET function (see SAM-2).
08h	The task manager shall perform a LOGICAL UNIT RESET function (see SAM-2).
20h	The task manager shall perform a TARGET RESET function (see SAM-2).
40h	The task manager shall perform a CLEAR ACA function (see SAM-2).
All other values reserved	The task manager shall return an SVP_RSP IU containing GOOD status. The RSP_CODE shall be set to TASK MANAGEMENT FUNCTION NOT SUP-PORTED.

The TAG OF TASK TO BE MANAGED field specifies the TAG value from the SVP_CMD information unit that contained the task to be managed (e.g. the task to be aborted by an ABORT TASK request). This field is reserved if the task management request is not directed to an I_T_L_Q nexus.