LSI LOGIC®

4420 ArrowsWest Drive
Colorado Springs, CO 80907

August 14, 2000

To: T10 Technical Committee
From: John Lohmeyer, LSI Logic Principal Member of T10
Subj: Expander Communication Protocol

Revision 1 changes:

1. Changed ECP protocol to be 8-bit asynchronous instead of 16-bit asynchronous.
2. Changed the USED FLAG from one byte to one bit and renamed it USED.
3. Added an address assignment function and defined new direct addressing functions that work with these addresses. This permits larger function structures and it allows functions to be sent to expanders that currently do not have any targets addressable on the far port.
4. Added an enabling mechanism so that legacy initiators do not accidentally issue ECP functions. Picked an eight-byte random number for the expander function signature.
5. Added an initiator data structure and a target data structure so that the same mechanism can be used to set and report initiator, expander, and target margin settings.
6. Added capabilities to enable, disable, reset, and force single-ended mode on the far port.
7. Added ability to report the target port's bus mode.
8. Added support for the SCSI INQUIRY command to expanders, including relevant parts of the standard INQUIRY data and VPD data.
9. Added rules for handling of reserved and unimplemented vendor-specific functions.
10. Removed the requirement to force bad parity on altered data if the incoming parity is bad.

I received a request to include the ability to report skew compensation and AAF settings. I have not yet done so, but expect that this could be done with an inbound direct-addressing expander function. While the values for these settings are likely to be vendor specific, I think we could standardize the number of fields and the size of the fields. I am especially looking for input on how big to make these fields.

I have documented the ECP proposal in the form of an annex for SPI-4 on the following pages.

Annex X

(normative)

Expander Communication Protocol

X.1 Introduction

This annex describes a method of expander communication and topology discovery called Expander Communication Protocol (ECP). This protocol permits initiators to detect expanders that support the protocol. It also permits the initiator to pass parameter settings to expanders and permits expanders to report settings and status information. No new hardware features are required of initiators or targets to implement this protocol.

ECP depends on the expander being able to monitor the data transfers associated with WRITE BUFFER and READ BUFFER commands (see SPC-2) and to alter specific portions of the data transferred as it passes through the expander. To simplify the expander implementation requirements, the ECP protocol is restricted to 8-bit asynchronous transfers.

X.2 Glossary and Definitions

X.2.1 Communicative expander: A simple expander (see Annex F) that has the additional capability to support the requirements of this annex and thus is capable of transmitting information beyond that received on its ports to specific other entities in the domain. In this annex, unless stated otherwise, the term expander means communicative expander.

X.2.2 Expander function signature: A specific sequence of data bytes that identifies an ECP command in the data phase of a WRITE BUFFER or READ BUFFER command.

X.2.3 Far port: For the current connection, an expander port that is not the near port.

X.2.4 First expander: In a series expander set, the expander that couples the bus segment containing the initiator to the next bus segment on the path to the target.

X.2.5 Last expander: In a series expander set, the expander that couples the bus segment containing the target to the next bus segment on the path to the initiator.

X.2.6 Near port: For the current connection, the expander port connected directly to the initiator through a bus segment or connected to the initiator through other expanders and bus segments.

X.2.7 Non-target port: A far port that is not a target port.

X.2.8 n^{th} expander: In a series expander set, the n^{th} expander on the path from the initiator to the target.

X.2.9 Path: The set of all bus segments and expanders between an initiator and a target.

X.2.10 Series expander set: The set of all expanders that couple the bus segment containing an initiator to the bus segment containing a target.

X.2.11 Target port: For the current connection, a far port that is connected directly to the target through a bus segment or connected to the target through other expanders and bus segments.

X.3 Enabling ECP Protocol

Following a power cycle or bus reset, a communicative expander shall function as a simple expander for each initiator until the initiator enables ECP protocol as follows:

1. Negotiate asynchronous transfer mode and a transfer width of 8 bits
2. Issue a WRITE BUFFER command to any target device with the MODE field set to Echo buffer plus enable ECP (1Ah).

The initiator may disable ECP protocol by:

1. Negotiate asynchronous transfer mode and a transfer width of 8 bits
2. Issue a WRITE BUFFER command to any target device with the MODE field bit set to Disable ECP (1Bh).

[Editor's Note: This proposal would expand the MODE field in the WRITE BUFFER command (SPC-2) from 4 bits to 5 bits adding the above two code values and reserving all other new code values. These values were selected so that enabling ECP and writing to the echo buffer can be combined in one operation with MODE 1Ah. Legacy targets may interpret this value as either Echo buffer (0Ah) or Write data (02h), which should cause no harm. Legacy targets may interpret the Disable ECP code value as either 0Bh or 03h, both of which are reserved and should cause no harm.**]**

This enabling and disabling of ECP protocol is done on an initiator basis (i.e., each initiator issues a single WRITE BUFFER command to enable or disable ECP protocol for that initiator for all communicative expanders on the bus).

Note: While the initiator has ECP protocol enabled, it is responsible for not issuing WRITE BUFFER commands with the EXPANDER FUNCTION SIGNATURE in the first 8 bytes of the data buffer unless it is an expander function header (see X.4).

X.4 Communicative expander function structures

Communicative expander functions consist of outbound and inbound functions. The outbound functions are contained in the data of a WRITE BUFFER command with the MODE field set to Write data (00010b), Echo buffer mode (01010b) or Echo buffer plus enable ECP mode (11010b). The inbound functions return information in the data of a READ BUFFER command with the mode field set to Data (00010b), Echo buffer mode (01010b) or Echo buffer plus enable ECP mode (11010b).

The outbound and inbound functions are further divided into relative and direct addressing functions. For relative addressing functions, the data is transferred in a 172-byte data structure consisting of a 16-byte expander function header followed by ten 16-byte control/data structures. For direct addressing functions, the data is transferred in a function-specific data structure, which follows the 16-byte expander function header. In either case, the first 16 bytes of the data structure contain an expander function header as shown in table X.1.

Table X.1 — Expander function header

Bit / Byte	7	6	5	4	3	2	1	0
0	(MSB)							
1								
2								
3				EXPANDER FUNCTION SIGNATURE				
4				(B73384B8508F27DEh)				
5								
6								
7								(LSB)
8	EXPANDER FUNCTION CODE							
9								
10								
11								
12				Function specific				
13								
14								
15								

The EXPANDER FUNCTION SIGNATURE contains a code of B73384B8508F27DEh that signifies this WRITE BUFFER or READ BUFFER data is an expander function. Any other code value is a normal WRITE BUFFER or READ BUFFER command and shall be repeated by communicative expanders but otherwise shall be ignored.

The EXPANDER FUNCTION CODES are documented in table X.2.

The function-specific bytes are not used for relative addressing functions and are reserved for future use. The function-specific bytes are documented for direct addressing functions in X.5.2 and X.5.4.

Table X.2 — Expander functions

EXPANDER FUNCTION CODE	Expander function	Type
00h	Reserved	Outbound Relative Addressing
01h	RESET ALL	
02h	RESET SPECIFIC	
03h	MARGIN CONTROL	
04h	ASSIGN ADDRESS	
05h - 2Fh	Reserved	
30h - 3Fh	Vendor specific	
40h	CONTROL	Outbound Direct Addressing
41h - 6Fh	Reserved	
70h - 7Fh	Vendor specific	
80h	Reserved	Inbound Relative Addressing
81h	REPORT CAPABILITIES	
82h	Reserved	
83h	MARGIN REPORT	
84h - AFh	Reserved	
B0h - BFh	Vendor specific	
C0h	EXPANDER INQUIRY	Inbound Direct Addressing
C1h - DFh	Reserved	
F0h - FFh	Vendor specific	

The outbound relative addressing expander functions are documented in X.5.1 and the outbound direct addressing expander functions are documented in X.5.2. The inbound relative addressing expander functions are documented in X.5.3 and the inbound direct addressing expander functions are documented in X.5.4.

For outbound and inbound relative addressing functions, the expander function header is followed by ten control/data structures as shown in table X.3. For outbound and inbound direct addressing functions, the expander function header is followed by one function-specific expander structure documented in X.5.2 and X.5.4.

Table X.3 — Control/data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	Reserved						
1	Function-specific fields							
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

The USED bit is documented in X.5.1.1, X.5.2.1, X.5.3.1, and X.5.4.1.

The remaining fields in the control/data structure are specific to the expander function and are documented in X.5.

X.5 Expander functions

X.5.1 Outbound relative addressing functions

X.5.1.1 Outbound relative addressing data transfer rules

Outbound relative addressing expander functions shall be performed during a WRITE BUFFER command with the MODE field set to one of the three values specified in X.4. The initiator shall transfer an expander function header followed by ten control/data structures. The first of the control/data structures shall have the USED bit set to 1 and may be used by the initiator to indicate its parameters or settings for the specified expander function. The initiator shall set the USED bit to 0 in the remaining nine control/data structures. The last control/data structure may be used to indicate the target's parameters or settings for the specified expander function.

Each expander in the domain shall repeat the entire data structure (expander function header plus ten control/data structures) without alteration to its non-target port or ports, if any. Each expander in the domain shall repeat the entire data structure without alteration to its target port, if any, except in the first control/data structure encountered with a USED bit of 0, it shall change the USED bit to 1. The expander shall interpret the other fields of this altered control/data structure as described below for the EXPANDER FUNCTION CODE.

An expander that receives a reserved or unimplemented vendor-specific relative addressing EXPANDER FUNCTION CODE shall follow all of the rules in this sub-clause, but shall ignore the contents of the altered control/data structure.

X.5.1.2 RESET ALL

The RESET ALL expander function resets all expanders in the domain to their default settings, the same as would be present at power up or following a reset condition. Ten control/data structures (table X.4) are transferred without alteration by all enabled expanders in the domain.

Table X.4 — RESET ALL control/data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	Reserved						
1	Reserved							
2								
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

X.5.1.3 RESET SPECIFIC

The RESET SPECIFIC expander function resets specified expanders to the default settings. The control/data structure for this expander function is shown in table X.5.

Table X.5 — RESET SPECIFIC control/data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	Reserved						
1	Reserved							R_EXP
2	Reserved							
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

A R_EXP bit of one indicates that the expander shall reset all of its settings to their default condition. A R_EXP bit of zero indicates that no settings shall be changed for this expander.

X.5.1.4 MARGIN CONTROL

The MARGIN CONTROL expander function sets various parameter settings in the initiator, expander, or target for usage between the initiator-target pair on subsequent synchronous and paced transfers. These parameter settings shall remain in effect until changed by another expander function (RESET ALL, RESET SPECIFIC with an R_EXP bit of 1, MARGIN CONTROL, or CONTROL with an R_EXP bit of 1) or by a reset condition.

The MARGIN CONTROL control/data structure is shown in table X.6

Table X.6 — MARGIN CONTROL control/data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	Reserved						
1	DRIVER STRENGTH (near port)				DRIVER TIMING SKEW (near port)			
2	SIGNAL GROUND BIAS (near port)				DRIVER PRECOMPENSATION (near port)			
3	SLEW RATE (near port)				Reserved			
4	Reserved				Reserved			
5	Reserved				Reserved			
6	Reserved				Reserved			
7	Reserved				Vendor specific (near port)			
8	Reserved							
9	DRIVER STRENGTH (target port)				DRIVER TIMING SKEW (target port)			
10	SIGNAL GROUND BIAS (target port)				DRIVER PRECOMPENSATION (target port)			
11	SLEW RATE (target port)				Reserved			
12	Reserved				Reserved			
13	Reserved				Reserved			
14	Reserved				Reserved			
15	Reserved				Vendor specific (target port)			

Two sets of margin control fields (DRIVER STRENGTH, DRIVER TIMING SKEW, SIGNAL GROUND BIAS, DRIVER PRECOMPENSATION, and SLEW RATE) are provided, one set for the near port and another set for the target port. The first control/data structure is used for the initiator settings, in which only the target port fields are used; the near port fields are reserved. The last control/data structure is used for the target settings, in which only the near port fields are used; the target port fields are reserved.

The margin control fields shall be implemented as two's-complement values with 0000b being the nominal value. The maximum supported setting for each field shall be 0111b and the minimum supported setting for each field shall be 1111b. Up to 16 distinct values are available for each field. Expanders that support fewer than 16 distinct values for a field should round intermediate settings to a supported value.

In the case of the SIGNAL GROUND BIAS fields, values 0000b through 0111b shall enable the bias cancellation circuit and values 1000b through 1111b shall disable the bias cancellation circuit, if disabling of this circuit is supported.

X.5.1.5 ASSIGN ADDRESS

The ASSIGN ADDRESS expander function is used to assign a direct address to one or more expanders. The control/data structure for this expander function is shown in table X.7.

Table X.7 — ASSIGN ADDRESS control/data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	Reserved						
1	ASSIGN	ADDRESS						
2	Reserved							
3								
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

The ASSIGN bit of 1 indicates that the expander shall respond to the address specified in the ADDRESS field for direct addressing expander functions. The address assignment shall remain in effect until changed by another ASSIGN ADDRESS function or until the next reset condition or power cycle. An ASSIGN bit of 0 indicates that the expander shall not change its address assignment.

X.5.2 Outbound direct addressing functions

X.5.2.1 Outbound direct addressing data transfer rules

Outbound direct addressing expander functions shall be performed during a WRITE BUFFER command with the MODE field set to one of the three values specified in X.4. The initiator shall transfer an expander function header followed by a function-specific data structure.

Each expander in the domain shall repeat the entire data structure (expander function header plus the function-specific data structure) without alteration to its non-target port or ports, if any. Each expander in the domain shall repeat the entire data structure without alteration to its target port, if any, except if the ADDRESS field in the first byte of the function-specific data structure matches its currently assigned address (see X.5.1.5) and the USED bit is 0, it shall change the USED bit to 1. The expander shall interpret the other fields of this altered function-specific data structure as described below for the EXPANDER FUNCTION CODE.

An expander that receives a reserved or unimplemented vendor-specific direct addressing EXPANDER FUNCTION CODE shall follow all of the rules in this sub-clause, but shall ignore the contents of the altered function-specific data structure.

X.5.2.2 CONTROL

The CONTROL function is used to set or clear parameters on the addressed expander. The function-specific bytes in the expander function header are reserved for this function.

Table X.8 — CONTROL data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	ADDRESS						
1	TARGET_ID				D_FAR	E_FAR	R_FAR	FSE_FAR
2	Reserved							R_EXP
3	Reserved							
4								
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								

The TARGET_ID field shall be set to the SCSI device address of a target on one of the expander's far ports. This identifies which far port is to be controlled by the D_FAR, E_FAR, R_FAR, or FSE_FAR bits on expanders that have multiple far ports.

A D_FAR bit of one indicates that the far port associated with TARGET_ID shall be disabled. That is, the expander shall not drive signals on that far port and shall ignore all signals on that far port. A D_FAR bit of zero does not request any control function.

An E_FAR bit of one indicates that the far port associated with TARGET_ID shall be enabled. That is, the expander shall resume expander functions for that far port. An E_FAR bit of zero does not request any control function.

A R_FAR bit of one indicates that the far port associated with TARGET_ID shall be reset by creating a reset condition on that port. A R_FAR bit of zero does not request any control function.

A FSE_FAR bit of one indicates that the port associated with TARGET_ID shall be forced into single ended mode (by placing the appropriate signal voltage on the DIFFSENS line). A FSE_FAR bit of zero does not request any control function.

A R_EXP bit of one indicates that the expander shall reset all of its settings to their default condition. A R_EXP bit of zero indicates that no settings shall be changed for this expander.

X.5.3 Inbound relative addressing functions

X.5.3.1 Inbound relative addressing data transfer rules

The initiator shall set the USED bit of the first control/data structure to 1 and shall set the used bit in the remaining nine control/data structures to 0. The data structure containing an inbound relative addressing function is then placed in the target's buffer using a WRITE BUFFER command with the MODE field set to one of the three values specified in X.4. Expanders shall not alter the data structure during the WRITE BUFFER command if the EXPANDER FUNCTION CODE is 80h to FFh. The inbound relative addressing function is then performed during a subsequent READ BUFFER command with the MODE field set to one of the three values specified in X.4.

Targets that implement the ECP should alter the first control/data structure to report capabilities or margin settings.

During the data transfer phase of the READ BUFFER command, each expander with a target port shall repeat the entire data structure without alteration to its near port, except the expander shall alter the first control/data structure encountered with a USED bit of 0 by changing the USED bit to 1 and altering the other fields of this control/data structure as described below for the EXPANDER FUNCTION CODE.

An expander that receives a reserved or unimplemented vendor-specific relative addressing EXPANDER FUNCTION CODE shall follow all of the rules in this sub-clause, but shall output 00h in bytes 1 through 15 of the control/data structure.

X.5.3.2 REPORT CAPABILITES

The REPORT CAPABILITES function is used to determine domain topology and report expander characteristics. The REPORT CAPABILITES control/data structure is shown in table X.9.

Table X.9 — REPORT CAPABILITES control/data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	Reserved						
1	(MSB)	FAR SCSI ID LIST						
2								(LSB)
3	MINIMUM TRANSFER PERIOD FACTOR							
4	MAXIMUM REQ/ACK OFFSET							
5	MAXIMUM TRANSFER WIDTH EXPONENT							
6	PROTOCOL OPTION BITS SUPPORTED							
7	PORTS				Reserved			TARG_MODE
8	Reserved							
9								
10								
11								
12								
13								
14								
15								

The FAR SCSI ID LIST field contains the inclusive OR of all SCSI IDs known to be located on the target port of the expander. For example, if SCSI devices with IDs 0, 1, and 12 were previously accessed on the target port, the expander sets this field to 1003h.

The MINIMUM TRANSFER PERIOD FACTOR field shall be set to the smallest value of the TRANSFER PERIOD FACTOR (see 16.3.10.1) supported by the expander.

The MAXIMUM REQ/ACK OFFSET field shall be set to the largest value of the REQ/ACK OFFSET (see 16.3.10.1) supported by the expander.

The MAXIMUM TRANSFER WIDTH EXPONENT field shall be set to the largest value of the TRANSFER WIDTH EXPONENT (see 16.3.10.1) supported by the expander.

The PROTOCOL OPTIONS BITS SUPPORTED field shall set the corresponding bit to one for each supported protocol option bit in byte 7 of the PPR message (see 16.3.10.1).

The PORTS field shall contain the number of ports on the expander not including the near port. A value of 0 in this field indicates that the expander is not capable of reporting this information.

The TARG_MODE field specifies the current bus mode for the target port as defined in table X.10.

Table X.10 — Far port bus mode

TARG_MODE	Target port bus mode
00b	Unknown (expander not capable of reporting bus mode)
01b	Single ended
10b	Low Voltage Differential
11b	High Voltage Differential

X.5.3.3 MARGIN REPORT

The MARGIN REPORT expander function is used to report the current margin settings for the initiator, expander, or target. The MARGIN REPORT control/data structure is shown in table X.11.

Table X.11 — MARGIN REPORT control/data structure

Bit / Byte	7	6	5	4	3	2	1	0
0	USED	Reserved						
1	DRIVER STRENGTH (near port)				DRIVER TIMING SKEW (near port)			
2	SIGNAL GROUND BIAS (near port)				DRIVER PRECOMPENSATION (near port)			
3	SLEW RATE (near port)				Reserved			
4	Reserved				Reserved			
5	Reserved				Reserved			
6	Reserved				Reserved			
7	Reserved				Vendor specific (near port)			
8	Reserved							
9	DRIVER STRENGTH (target port)				DRIVER TIMING SKEW (target port)			
10	SIGNAL GROUND BIAS (target port)				DRIVER PRECOMPENSATION (target port)			
11	SLEW RATE (target port)				Reserved			
12	Reserved				Reserved			
13	Reserved				Reserved			
14	Reserved				Reserved			
15	Reserved				Vendor specific (target port)			

Two sets of margin report fields (DRIVER STRENGTH, DRIVER TIMING SKEW, SIGNAL GROUND BIAS, DRIVER PRECOMPENSATION, and SLEW RATE) are provided, one set for the near port and another set for the target port. The first control/data structure is used for the target settings, in which only the near port fields are used; the target port fields are reserved. The last control/data structure is used for the initiator settings, in which only the target port fields are used; the near port fields are reserved. The initiator may or may not set the USED bit to 1 in the last control/data structure.

The margin report fields shall return the current settings for the initiator-target pair. Fields that are not implemented shall be reported as 0000b. Otherwise, the current setting for the field, possibly rounded as described in X.5.1.3, shall be returned.

X.5.4 Inbound direct addressing functions

X.5.4.1 Inbound direct addressing data transfer rules

The initiator shall set the USED bit in the function-specific data structure to 0. The data structure containing an inbound direct addressing function is then placed in the target's buffer using a WRITE BUFFER command with the MODE field set to one of the three values specified in X.4. Expanders shall not alter the data structure during the WRITE BUFFER command if the EXPANDER FUNCTION CODE is 80h to FFh. The inbound direct addressing expander function is then performed during a subsequent READ BUFFER command with the MODE field set to one of the three values specified in X.4.

During the data transfer phase of the READ BUFFER command, each expander with a target port shall repeat the entire data structure without alteration to its near port, except if the address field in the first byte of the function-specific data structure matches its currently assigned address (see X.5.1.5) and the USED bit is 0, it shall change the USED bit to 1. The expander shall alter the other fields of this function-specific data structure as described below for the EXPANDER FUNCTION CODE.

An expander that receives a reserved or unimplemented vendor-specific direct addressing EXPANDER FUNCTION CODE shall follow all of the rules in this sub-clause, but shall output 00h in remaining bytes of the data structure.

X.5.4.2 EXPANDER INQUIRY

The EXPANDER INQUIRY function is used to report information about the specified expander in a manner similar to the SCSI INQUIRY command documented in SPC-2. The expander function header for this function shall include function specific fields as described in table X.12.

Table X.12 — Expander INQUIRY function header

Bit / Byte	7	6	5	4	3	2	1	0
0	(MSB)							
1								
2								
3				EXPANDER FUNCTION SIGNATURE (B73384B8508F27DEh)				
4								
5								
6								
7								(LSB)
8	EXPANDER INQUIRY FUNCTION CODE (C0h)							
9	Reserved							EVPD
10	PAGE CODE							
11	Reserved							
12	ALLOCATION LENGTH							
13								
14				Reserved				
15								

An enable vital product data (EVPD) bit of one specifies that the expander shall return the optional vital product data specified by the PAGE CODE field (see the vital product data parameters documentation in SPC-2). If the expander does not support the optional vital product data, then it shall return all zero bytes for the specified allocation length. If the EVPD bit is zero, then the expander shall return Expander INQUIRY data as documented in table X.13.

Table X.13 — Expander INQUIRY data

Bit / Byte	7	6	5	4	3	2	1	0
0								
1				Reserved				
2								
3								
4	ADDITIONAL LENGTH (33h)							
5								
6				Reserved				
7								
8	(MSB)							
15				VENDOR IDENTIFICATION				(LSB)
16	(MSB)							
31				PRODUCT IDENTIFICATION				(LSB)
32	(MSB)							
35				PRODUCT REVISION LEVEL				(LSB)
36								
55				Vendor-specific				

The VENDOR IDENTIFICATION, PRODUCT IDENTIFICATION, and PRODUCT REVISION LEVEL fields shall return data as documented in SPC-2 for the Standard INQUIRY data format.

X.6 Data Transfer Requirements

The communicative expander functions shall only be performed when the data transfer agreement is 8-bit asynchronous. For any other data transfer agreement, the communicative expander shall operate as a simple expander.

When altering data, communicative expanders shall construct correct parity for the altered data on the outgoing port.