RapidIO
Interconnect Specification
Device Compliance Checklists

Rev. X.Y.Z MM/20YY

Revision History

Revision	Description	Date
1.0	First release	MM/DD/YYYY

Questions regarding the RapidIO Trade Association, specifications, or membership should be forwarded to:
Suite 325, 3925 W. Braker Lane
Austin, TX 78759
512-305-0070 Tel.
512-305-0009 FAX.

Chapter 1 Overview

Introduction

The document contains the device compliance checklists adhering to Part 6: LP-Serial Physical Layer Specification of the RapidIO Interconnect Specification revision 2.2.1. If an inconsistency exists between the specifications and the checklists defined by this document, the specifications take precedence.

References

- Part 6: LP-Serial Physical Layer Specification, Revision 2.2.1, MM/YYYY

Item #	Compliance Item	Specification Reference	Optional	Interop Item
2	Packets	Part 6, Chap. 2		
2.2	Packet Field Definitions	Part 6, Sec. 2.2		
2.2A	A sRIO packet should have all required field as specified in Part 6 sec 2.2 Table 2-1.			Inter-op
2.2B	Acknowledge ID (ackID) is the packet identifier for link level packet acknowledgment. When the short control symbol is being used, the ackID value shall be 5 bits long and shall be left justified in the ackID field (ackID[0-4]) with the right most bit of the field (ackID[5]) set to 0b0. When the long control symbol is being used, the ackID value shall be 6 bits long which fills the ackID field.			
2.2C	If VC=0 and Critical Request Flow is not supported, the CRF bit is reserved.			
2.2D	The usage of VC, PRIO and CRF fields in a packet shall follow Part 6, Table 2-2.			
2.3	Packet Format	Part 6, Sec. 2.3		
2.3A	the physical layer ackID, VC, CRF, and prio fields are prefixed at the beginning of the packet and a 16-bit CRC field is appended to the end of the packet.			Inter-op
2.3B	Packet format shall follow Part 6, Figure 2-1.			
2.3C	LP-Serial packets shall have a length that is an integer multiple of 32 bits.			Inter-op
2.3D	If the length of a packet is an odd multiple of 16 bits, a 16-bit pad whose value is 0 (0x0000) shall be appended at the end of the packet.			Inter-op
2.4	Packet Protection	Part 6, Sec. 2.4		
2.4A	The 16-bit CRC covers the entire packet except for the ackID field, which is considered to be zero for the CRC calculations.			Inter-op
2.4.1A	For a packet whose length, exclusive of CRC, is 80 bytes or less, a single CRC is appended at the end of the logical fields.			Inter-op
2.4.1B	For packets whose length, exclusive of CRC, is greater than 80 bytes, a CRC is added after the first 80 bytes and a second CRC is appended at the end of the logical layer fields.			Inter-op
2.4.1C	The second CRC value is a continuation of the first. The first CRC is included in the running calculation, meaning that the running CRC value is not re-initialized after it is inserted after the first 80 bytes of the packet.			Inter-op
2.4.1D	If the CRC appended to the end of the logical layer fields does not cause the end of the resulting packet to align to a 32-bit boundary, a two byte pad of all logic 0s is postpended to the packet.			
2.4.2	CRC-16 Code	Part 6, Sec. 2.4.2		
2.4.2A	The ITU polynomial X16+X12+X5+1 shall be used to generate the 16-bit CRC for packets.			
2.4.2B	The value of the CRC shall be initialized to 0xFFFF (all logic 1s) at the beginning of each packet.			Inter-op
2.5	Maximum Packet Size	Part 6, Sec. 2.5		
2.5A	The maximum transmitted packet size permitted by the LP-Serial specification is 276 bytes.			Inter-op

Item #	Compliance Item	Specification Reference	Optional	Interop Item
3	Control Symbols	Part 6, Chap. 3		
3.1	Introduction	Part 6, Sec. 3.1		
3.1A	For forward compatibility, a control symbol function received by a port with an encoding in one or more of the fields assigned to the function that the port does not understand or support shall be handled as follows. If an encoding that the port does not understand or support occurs in a functional field, the control symbol function shall be ignored. If an encoding that the port does not understand or support occurs only in an informational field, the control symbol function shall be executed. In either case, no error shall be reported.	Part 6, Sec. 3.1		Inter-op
3.2	Control Symbol Field Definitions			
3.2A	Control symbol fields shall be implemented as defined in Part 6, Table 3-1.			Inter-op
3.3	Control Symbol Format	Part 6, Sec. 3.3		
3.3A	Short control symbols shall have the 24 data bit format shown in Part 6, Figure 3-1.			Inter-op
3.3B	All long control symbols shall have the 48 data bit format shown in Part 6, Figure 3-2.			Inter-op
3.3C	The receiver shall support control symbols with two functions so that a packet acknowledgment and a packet delimiter can be carried in the same control symbol.			Inter-op
3.4	Stype0 Control Symbols	Part 6, Sec. 3.4		
3.4A	The encoding and function of stype0 and the information carried in parameter0 and parameter1 for each stype0 encoding shall be as specified in Part 6, Table 3-2.			
3.4B	“Status” (0b100) is the default stype0 encoding and is used when a control symbol does not convey another stype0 function.			Inter-op
3.4C	Stype0 parameter shall be implemented as defined in Part6, Table 3-3.			
3.4D	ackID_status parameter indicates the value of the ackID field expected in the next packet the port receives. This value is 1 greater than the ackID of the last packet accepted by the port exclusive of CT mode packets accepted after the port entered an Input-stopped state.			Inter-op
3.4E	The value of the buf_status field in a packet-accepted control symbol is inclusive of the receive buffer consumption of the packet being accepted.			Inter-op
3.4F	For short control symbols, buf_status=0-30 indicates the number of maximum sized packet buffers the port has available for reception on the specified VC. Buf_statuss=31 indicates the port has an undefined number of maximum sized packet buffers available for packet reception, and relies on retry for flow control.			

3.4G	For long control symbols, buf_status=0-62 indicates the number of maximum sized packet buffers the port has available for reception on the specified VC. Buf_statuss=63 indicates the port has an undefined number of maximum sized packet buffers available for packet reception, and relies on retry for flow control.			
3.4.1	Packet-Accepted Control Symbol	Part 6, Sec. 3.4.1		
3.4.1A	The packet-accepted control symbol formats shall follow Part 6, Figure 3-3.			
3.4.1B	The packet-accepted control symbol indicates that the port sending the control symbol has taken responsibility for sending the packet to its final destination and that resources allocated to the packet by the port receiving the control symbol can be released.			
3.4.1C	The packet-accepted control symbol shall be generated only after the entire packet has been received and found to be free of detectable errors.			Inter-op
3.4.1D	The port receiving the packet-accepted control symbol must reassociate the ackID in the packet_ackID field with the VC of the accepted packet to determine the VC to which the buf_status applies.			Inter-op
3.4.2	Packet-Retry Control Symbol	Part 6, Sec. 3.4.2		
3.4.2A	The packet-retry control symbol format shall follow Part 6, Figure 3-4			
3.4.2B	A packet-retry control symbol indicates that the port sending the control symbol was not able to accept the packet due to some temporary resource conflict such as insufficient buffering and the packet must be retransmitted.			
3.4.2C	The packet-retry control symbol is only used in singleVC mode for compatibility with Rev. 1.x RapidIO devices. Packet retry is replaced with error recovery when multiple VCs are active.			Inter-op
3.4.3	Packet-Not-Accepted Control Symbol	Part 6, Sec. 3.4.3		
3.4.3A	The packet-not-accepted control symbol format shall follow Part 6, Figure 3-5.			
3.4.3B	The packet-not-accepted control symbol indicates that the port sending the control symbol has either detected an error in the received character stream or, when operating in multiple VC mode, has insufficient buffer resources and as a result may have rejected a packet or control symbol.			
3.4.3C	The packet-not-accepted control symbol cause field shall be encoded as specified in Table 3-4 of Part6. If the port issuing the control symbol is not able to specify the fault, or the fault is not one of those listed in the table, the general error encoding shall be used.			
3.4.4	Status Control Symbol	Part 6, Sec. 3.4.4		
3.4.4A	The status control symbol format shall follow Part 6, Figure 3-6.			
3.4.4B	The status control symbol indicates receive status information (ackID_status and buf_status) about the port sending the control symbol.			
3.4.5	VC-Status Control Symbol	Part 6, Sec. 3.4.5		

3.4.5A	The VC-status control symbol format shall follow Part 6, Figure 3-7.			
3.4.5B	The VC-status control symbol indicates to the receiving port the available buffer space that the sending port has available for packet reception on the virtual channel (VC) specified in the control symbol.			
3.4.5C	VCID is 3-bit field that is right justified in the Parameter0 field of the control symbol. The remaining bits of the parameter0 field are reserved, set to 0 on transmission and ignored on reception.			
3.4.5D	The VCID field shall be implemented as Part 6, Table 3-5.		Required in multi-vc mode	
3.4.5E	The VC-status control symbol is used only for virtual channels 1 through 8 (VC1 through VC8) and may be transmitted only when the specified VC is implemented and enabled.			Inter-op
3.4.5F	The VC-status control symbol may be transmitted at any time and should be transmitted whenever the number of maximum length packet buffers available for reception on a VC has changed and has not been otherwise communicated to the connected port.			Inter-op
3.4.6	Link-Response Control Symbol	Part 6, Sec. 3.4.6		
3.4.6A	The link-response control symbol format shall follow Part 6, Figure 3-8.			
3.4.6B	The link-response control symbol is used by a port to respond to a link-request control symbol as described in the link maintenance protocol described in Section 5.7, "Link Maintenance Protocol".			
3.4.6C	The status reported in the link-response control symbol status field is the status of the port at the time the associated input-status link-request control symbol was received and is informational only.			Inter-op
3.4.6D	The encoding of the link-response control symbol port_status field shall follow Part 6, Table 3-6.			
3.5	Stype1 Control Symbols	Part 6, Sec. 3.5		
3.5A	The encoding of stype1 and the function of the cmd field shall follow Part 6, Table 3-7.			
3.5B	Restart-from-retry and link-request control symbols may only be packet delimiters if a packet is in progress.			
3.5C	NOP (Ignore) is the default value when the control symbol does not convey another stype1 function.			
3.5.1	Start-of-Packet Control Symbol	Part 6, Sec. 3.5.1		
3.5.1A	The start-of-packet control symbol is used to delimit the beginning of a packet.			
3.5.1B	The control symbol formats shall follow Part 6, Figure 3-9.			
3.5.2	Stomp Control Symbol	Part 6, Sec. 3.5.2		
3.5.2A	The stomp control symbol is used to cancel a partially transmitted packet.			
3.5.2B	The stomp control symbol formats shall follow Part 6, Figure 3-10.			
3.5.3	End-of-Packet Control Symbol	Part 6, Sec. 3.5.3		

3.5.3A	The end-of-packet control symbol is used to delimit the end of a packet.			
3.5.3B	The control symbol formats shall follow Part 6, Figure 3-11.			
3.5.4	Restart-From-Retry Control Symbol	Part 6, Sec. 3.5.4		
3.5.4A	This Restart-From-Retry control symbol is used to mark the beginning of packet retransmission, so that the receiver knows when to start accepting packets after the receiver has requested a packet to be retried.			
3.5.4B	The restart-from-retry control symbol cancels a current packet and may also be transmitted on an idle link.			
3.5.4C	The restart-from-retry control symbol format shall follow Part 6, Figure 3-12.			
3.5.5	Link-Request Control Symbol	Part 6, Sec. 3.5.5		
3.5.5A	A link-request control symbol is used by a port to either issue a command to the connected port or request its input port status.			
3.5.5B	A link-request control symbol always cancels a packet whose transmission is in progress and can also be sent between packets.			
3.5.5C	The link-request control symbol formats shall follow Part 6, Figure 3-13.			
3.5.5D	The link-request control symbol cmd field shall be implemented as defined in Part 6, Table 3-8.			
3.5.5.1	Reset-Device Command	Part 6, Sec. 3.5.5.1		
3.5.5.1A	The link-request reset-device command causes the receiving device to go through its reset or power-up sequence.			
3.5.5.1B	The reset-device command does not generate a link-response control symbol.			
3.5.5.1C	A port receiving a reset-device command in a link-request control symbol shall not perform the reset function unless it has received four reset-device commands in a row without any other intervening packets or control symbols, except status control symbols.			
3.5.5.2	Input-Status Command	Part 6, Sec. 3.5.5.2		
3.5.5.2A	The input-status command requests the receiving port to return a link-response control symbol containing the ackID value it expects to next receive on its input port and the current input port operational status for informational purposes.			
3.5.5.2B	link-request/input-status command causes the receiver to flush its output port of all control symbols generated by packets received before the input-status command.			
3.5.6	Multicast-Event Control Symbol	Part 6, Sec. 3.5.6		
3.5.6A	The multicast-event control symbol formats shall follow Part 6, Figure 3-14.			
3.6	Control Symbol Protection	Part 6, Sec. 3.6		
3.6A	A 5-bit CRC is used for the short control symbol. It should be generated according to the polynomial specified in Part 6, Sec 3.6.1.			
3.6B	A 13-bit CRC is used for the long control symbol. It should be generated according to the polynomial specified in Part 6, Sec 3.6.3.			
3.6C	For both 5-bit CRC and 13-bit CRC, they should be set to all 1's before comptutation.			

3.6.1	CRC-5 Code	Part 6, Sec. 3.6.1		
3.6.1A	The ITU polynomial X5+X4+X2+1 shall be used to generate the 5-bit CRC for short control symbols.			
3.6.1B	The 5-bit CRC shall be computed over 20 bits comprised of control symbol bits 0 through 18 plus a 20th bit that is appended after bit 18 of the control symbol. The added bit shall be set to logic 0 (0b0).			
3.6.1C	The CRC shall be computed beginning with control symbol bit 0.			
3.6.1D	Before the CRC is computed, the CRC shall be set to all 1’s (0b11111).			
3.6.1E	The CRC check bits c[0:4] shall occupy short control symbol bits [19:23] respectively.			
3.6.1F	The 5-bit CRC shall be generated by each transmitter and verified by each receiver using the short control symbol.			
3.6.3	CRC-13 Code	Part 6, Sec. 3.6.3		
3.6.3A	The polynomial x13 + x10 + x8 + x5 + x2 + 1 shall be used to generate the 13-bit CRC for long control symbols.			
3.6.3B	The 13-bit CRC shall be computed over control symbol bits 0 through 34 beginning with control symbol bit 0.			
3.6.3C	Before the 13-bit CRC is computed, the CRC shall be set to all 0’s (0b0_0000_0000_0000).			
3.6.3D	The CRC check bits c[0:12] shall occupy long control symbol bits [35:47] respectively.			
3.6.3E	The 13-bit CRC shall be generated by each transmitter and verified by each receiver using the long control symbol.			

Item #	Compliance Item	Specification Reference	Optional	Interop Item
4	PCS and PMA Layers	Part 6, Chap. 4		
4.5	8B/10B Transmission Code	Part 6 Sec. 4.5		
4.5.2	Running Disparity	Part 6 Sec. 4.5.2		
4.5.2A	The encoder and decoder each have a running disparity variable for each lane which are all independent of each other.			
4.5.2B	The current value of encoder running disparity is used to select which unbalanced code-group will be used when the encoding for a character requires a choice between two unbalanced code-groups.			
4.5.3	Running Disparity Rules	Part 6 Sec. 4.5.3		
4.5.3A	After power-up and before the port is operational, both the transmitter(encoder) and receiver (decoder) must establish current values of running disparity.			
4.5.3B	The transmitter shall use a negative value as the initial value for the running disparity for each lane. The receiver may use either a negative or positive initial value of running disparity for each lane.			
4.5.3C	The following algorithm shall be used for calculating the running disparity for each lane.			
4.5.3D	In the encoder, the algorithm operates on the code-group that has just been generated by the encoder. In the receiver, the algorithm operates on the received code-group that has just been decoded by the decoder.			
4.5.3E	Each code-group is divided to two sub-blocks where the first six bits (abcdei) form one sub-block (6-bit sub-block) and the second four bits (fghj) form a second sub-block (4-bit sub-block). Running disparity at the beginning of the 6-bit sub-block is the running disparity at the end of the preceding code-group. Running disparity at the beginning of the 4-bit sub-block is the running disparity at the end of the preceding 6-bit sub-block. Running disparity at the end of the code-group is the running disparity at the end of the 4-bit sub-block.			
4.5.3F	The sub-block running disparity shall be calculated as follows:			
4.5.3G	The running disparity is positive at the end of any sub-block if the sub-block contains more 1s than 0s. It is also positive at the end of a 4-bit sub-block if the sub-block has the value 0b0011 and at the end of a 6-bit sub-block if the sub-block has the value 0b000111.			
4.5.3H	The running disparity is negative at the end of any sub-block if the sub-block contains more 0s than 1s. It is also negative at the end of a 4-bit sub-block if the sub-block has the value 0b1100 and at the end of a 6-bit sub-block if the sub-block has the value 0b111000.			
4.5.3I	In all other cases, the value of the running disparity at the end of the sub-block is running disparity at the beginning of the sub-block (the running disparity is unchanged).			

4.5.4	8B/10B Encoding	Part 6 Sec. 4.5.4		
4.5.4A	When encoding a character, the code-group in the RD- column is selected if the current value of encoder running disparity is negative and the code-group in the RD+ column is selected if the current value of encoder running disparity is positive.			
4.5.4B	Data characters (Dx.y) shall be encoded according to Table 4-1 and the current value of encoder running disparity.			
4.5.4C	Special characters (Kx.y) shall be encoded according to Table 4-2 and the current value of encoder running disparity.			
4.5.4D	After each character is encoded, the resulting code-group shall be used by the encoder to update the running disparity according to the rules in Section 4.5.3, "Running Disparity Rules".			
4.5.5	Transmission Order.	Part 6 Sec. 4.5.5		Inter-op
4.5.5A	The parallel 10-bit code-group output of the encoder shall be serialized and transmitted with bit "a" transmitted first and a bit ordering of "abcdeifghj".			
4.5.6	8B/10B Decoding	Part 6 Sec. 4.5.6		
4.5.6A	The 8B/10B decoding function decodes received 10-bit code-groups into 9-bit characters and detects and reports received code-groups that have no defined decoding due to one or more transmission errors.			
4.5.6B	The decoding function uses Table 4-1, Table 4-2 and the current value of the decoder running disparity.			
4.5.6C	To decode a received code-group, the decoder shall select the RD- column of Table 4-1 and Table 4-2 if the current value of the decoder running disparity is negative or shall select the RD+ column if the value is positive.			
4.5.6D	The decoder shall then compare the received code-group with the code-groups in the selected column of both tables. If a match is found in one of the tables, the code-group is defined to be a "valid" code-group and is decoded to the associated character. If no match is found, the code-group is defined to be an "invalid" code-group and is decoded to a character that is flagged in some manner as INVALID.			
4.5.6E	After each code-group is decoded, the decoded code-group shall be used by the decoder to update the decoder running disparity according to the rules in Section 4.5.3, "Running Disparity Rules".			
4.5.6F	A comma is a pattern of 7 bits that is used by receivers to acquire code-group boundary alignment.			
4.5.6G	Two commas patterns are defined, 0b0011111 (comma+) and 0b1100000 (comma-). The pattern occurs in bits abcdeif of the special characters K28.1, K28.5 and K28.7.			

4.5.6H	Within the code-group set, it is a singular bit pattern, which, in the absence of transmission errors, cannot appear in any other location of a code-group and cannot be generated across the boundaries of any two adjacent code-groups with the following exception.			
4.5.6I	The /K28.7/ special code-group when followed by any of the data code-groups /D3.y/, /D11.y/, /D12.y/, /D19.y/, /D20.y/, /D28.y/, or /K28.y/, where y is an integer in the range 0 through 7, may (depending on the value of running disparity) cause a comma to be generated across the boundary of the two code-groups.			
4.5.6J	A comma that is generated across the boundary between two adjacent code-groups may cause the receiver to change the 10-bit code-group alignment. As a result, the /K28.7/ special code-group may be used for test and diagnostic purposes only.			
4.5.7	Special Characters and Columns	Part 6 Sec. 4.5.7		
4.5.7A	Table 4-3 defines the special characters and columns of special characters used by LP-Serial links.			
4.5.7B	A special character and its associated code-group that is defined by the 8B/10B code, but not specified for use by the LP-Serial protocol are declared to be an “illegal” character and “illegal” code-group respectively.			
4.5.7C	The special characters K23.7, K28.2, K28.4, K28.6, K28.7 and K30.7 are illegal characters, and if a link is operating with Idle Sequence 1, K28.1 is also an illegal character.			
4.5.8	Effect of Single Bit Code-Group Errors	Part 6 Sec. 4.5.8		
4.5.8A	Port detects all single bit code group errors as defined in Part 6, Section 4.5.8, Table 4-4.		Only required of links operating with the IDLE 1 sequence and short control symbols.	
4.6	LP-Serial Link Widths	Part 6, Sec. 4.6		
4.6A	All LP-Serial ports shall support operation on links with one lane per direction (1x mode) and may optionally support operation over links with 2, 4, 8 and/or 16 lanes per direction (respectively 2x mode, 4x mode, 8x mode and 16x mode).			Inter-op
4.6B	LP-Serial ports that support operation over two or more lanes per direction shall support 1x mode operation over two of those lanes, lane 0 and lane R (the redunancy lane).			Inter-op
4.6C	If the port supports operation over more than two (2x mode), lane R shall be lane 1.			Inter-op
4.6D	If the port supports operation over more than two lanes, lane R shall be lane 2.			Inter-op
4.7	Idle Sequence	Part 6, Sec. 4.7		
4.7A	When idle is transmitted by a LP-Serial port, an idle sequence shall be transmitted on each of the port’s active output lanes.			Inter-Op
4.7B	Ports operating in Nx mode shall not stripe the idle sequence across the active lanes; there is an idle sequence for each of the N lanes.			Inter-Op

4.7C	An uninitialized LP-Serial port (state variable port_initialized not asserted) shall continuously transmit an idle sequence on all active output lanes.			Inter-Op
4.7D	An initialized LP-Serial port (state variable port_initialized asserted) shall transmit an idle sequence on each of its active output lanes when there is nothing else to transmit.			Inter-Op
4.7E	On links operating in 1x mode, the first code-group of the idle sequence shall immediately follow the last code-group of the preceding control symbol.			Inter-Op
4.7F	When a link is operating in Nx mode, the first column of N idle code-groups shall immediately follow the column containing the last code-groups of the preceding control symbol.			Inter-Op
4.7.1	Clock Compensation Sequence	Part 6, Sec. 4.7.1		
4.7.1A	The “clock compensation sequence” is four character sequence comprised of a K special character immediately followed by three R special characters (K,R,R,R).			
4.7.1B	A port shall transmit a clock compensation sequence on each of its active output lanes at least once every 5000 characters transmitted per lane by the port.			
4.7.1C	When a clock compensation sequence is transmitted, the entire 4 character sequence shall be transmitted.			
4.7.1D	When transmitted by a port operating in Nx mode, the clock compensation sequence shall be transmitted in parallel on all N lanes resulting in the column sequence ||K||R||R||R||.			
4.7.2	Idle Sequence 1 (IDLE1)	Part 6, Sec. 4.7.2		
4.7.2A	The IDLE1 sequence shall comply with the following requirements:			
4.7.2A1	Each instance of an IDLE1 sequence shall begin with the K special character.			
4.7.2A2	The second, third and fourth characters of each IDLE1 sequence may be the R special character.			
4.7.2A3	Except when generating the clock compensation sequence, all characters following the first character of an IDLE1 shall be a randomly selected sequence of A, K and R special characters that is based on a pseudo-random sequence generator of 7th degree or greater and subject to minimum and maximum requirements on the spacing of the A special characters.			
4.7.2A4	The number of non-A special characters between A special characters within an IDLE1 sequence shall be no less than 16 and no more than 31.			
4.7.2A5	The number shall be pseudo-randomly selected based on a pseudo-random sequence generator of 7th degree of greater.			
4.7.2A6	The requirement on the number of characters between successive A special characters should be maintained between successive IDLE1 sequences to ensure that two successive A special characters are always separated by at least 16 non-A characters.			

4.7.2A7	Except when transmitting a clock compensation sequence, an IDLE1 sequence may be of any length and may be terminated after any code-group.			
4.7.2A8	Each instance of IDLE1 shall be a new IDLE1 sequence that is unrelated to any previous IDLE1 sequence. Once transmission of an IDLE1 sequence has begun, the sequence may only be terminated. It may not be interrupted or stalled and then continued later.			
4.7.2A9	When a port transmitting IDLE1 is operating in Nx mode, the port shall transmit the identical sequence of A, K and R special characters in parallel on each of the N lanes and the N idle sequences shall be aligned across the lanes such that the initial /K/ of the N sequences shall all occur in the same column and the last code-group of the N sequences shall all occur in the same column.			
4.7.4	Idle Sequence 2 (IDLE2)	Part 6, Sec. 4.7.4		
4.7.4A	The IDLE sequence 2 shall be comprised of a continuous sequence of idle frames and clock compensation sequences.			
4.7.4B	The minimum clock compensation sequence density (clock compensation sequences per characters transmitted per lane) shall comply with the requirements specified in Section 4.7.1, "Clock Compensation Sequence".			
4.7.4C	Each clock compensation sequence shall be followed by an idle frame.			
4.7.4D	Each idle frame shall be followed by either a clock compensation sequence or another idle frame.			
4.7.4E	When a port is operating in Nx mode, the sequence of clock compensation sequences and idle frames shall be the same for all N lanes.			
4.7.4F	After a port using IDLE2 is initialized (the port initialization state variable port_initialized is asserted), the port may terminate an IDLE2 sequence after any character of an idle frame to transmit a control symbol or a SYNC sequence immediately followed by a link-request control symbol subject to the following requirementsrequest control symbol subject to the following requirements:			
4.7.4F1	Each M special character transmitted that is part of the idle frame random data field shall be followed by a minimum of four (4) random data field random data characters.			
4.7.4F2	The sequence of four (4) M special characters at the beginning of a CS field marker shall not be truncated.			
4.7.4F3	A port operating in Nx mode shall terminate an IDLE2 sequence at exactly the same character position in the sequence for each of the N lanes.			
4.7.4G	Each instance of IDLE2 shall be a new IDLE2 sequence that is unrelated to any previous IDLE2 sequence. Once transmission of an IDLE2 sequence has begun, the sequence may only be terminated. It may not be interrupted or stalled and then continued later.			

4.7.4H	When a port transmitting IDLE2 is operating in Nx mode, the port shall transmit IDLE2 sequences in parallel on each of the N lanes.			
4.7.4I	The IDLE2 sequences transmitted on each of the N lanes shall be aligned across the lanes such that the first character of the N idle sequences shall all occur in the same column and the last character of the N idle sequences shall all occur in the same column.			
4.7.4.1	Idle Frame	Part 6, Sec. 4.7.4.1		
4.7.4.1A	Each idle frame shall be composed of three parts, a random data field, a command and status (CS) field marker and an encoded CS field as shown in Figure 4-5.			
4.7.4.1.1	Idle Sequence 2 Random Data Field	Part 6, Sec. 4.7.4.1.1		
4.7.4.1.1A	The IDLE2 random data field shall contain pseudo-random data characters and the A and M special characters.			
4.7.4.1.1B	The total length of the random data field shall be no less than 509 and no more than 515 characters.			
4.7.4.1.1C	Unless otherwise specified, the characters comprising the random data field shall be pseudo-random data characters.			
4.7.4.1.1D	The random data field of an idle frame that immediately follows a clock compensation sequence shall begin with a M special character.			
4.7.4.1.1E	Otherwise, the random data field of an idle frame shall begin with a pseudo-random data character.			
4.7.4.1.1F	Unless otherwise specified, the pseudo-random data characters in the random data field shall occur in contiguous sequences of not less than 16 and no more than 31 pseudo-random characters. The length of each contiguous sequence shall be pseudo-randomly selected.			
4.7.4.1.1G	Adjacent contiguous sequences shall be separated by a single A or M special character. Each separator shall be pseudo-randomly selected.			
4.7.4.1.1H	The last four (4) characters of the random data field shall be pseudo-random data characters.			
4.7.4.1.1I	The length of the first contiguous sequence of pseudo-random characters in the random data field shall be no less than 16 and no more than 35 characters.			
4.7.4.1.1J	The length of last contiguous sequence of pseudo-random characters in the random data field shall be no less than 4 and no more than 35 characters.			
4.7.4.1.1K	Each random data field that is transmitted on a given lane of a link shall be generated by first generating a prototype random data field using the above rules, but with a D0.0 character in the place of each pseudo-random data character, and then scrambling the prototype random data field with the transmit scrambler for that lane.			
4.7.4.1.1L	The scrambling shall be done in exactly the same manner as packet and control symbol data characters are scrambled.	Part 6, Sec. 4.8.1		

4.7.4.1.1M	When a port is operating in Nx mode, the location A or M special characters in a random data field shall be identical for all N lanes.			
4.7.4.1.2	Idle Sequence 2 CS Field Marker	Part 6, Sec. 4.7.4.1.2		
4.7.4.1.2A	The CS field marker shall be the 8 character sequence M, M, M, M, D21.5, Dx.y, D21.5, !Dx.y as described in 4.7.4.1.2.			
4.7.4.1.2B	As shown above, the CS frame marker characters shall be transmitted from left to right. The first character transmitted is M, the last character transmitted is !Dx.y.			
4.7.4.1.2C	The active_port_width field shall be encoded as specified in Table 4-5.			
4.7.4.1.2D	The lane_number field shall be encoded as specified in Table 4-6.			
4.7.4.1.2E	A CS field marker whose first four characters are not all M special characters, fifth and seventh characters are not both D21.5 or D10.2 or sixth and eight character are not the bit wise complements of each other shall be determined to be corrupted.			
4.7.4.1.2F	A received CS field marker that is determined to be truncated and/or corrupted shall be ignored and discarded.			
4.7.4.1.2G	Any error detected in a truncated and/or corrupted CS field marker that is determined to be the result of a transmission error and not the result of truncation, such as an “invalid” or “illegal” character, shall be reported as an input error.			
4.7.4.1.3	IDLE2 Command and Status Field (CS field)	Part 6, Sec. 4.7.4.1.3		
4.7.4.1.3A	The CS field shall have 32 information bits, cs_field[0-31], and 32 check bits, cs_field[32-63].			
4.7.4.1.3B	The check bits cs_field[32-63] shall be the bit wise complement of the information bits cs_field[0-31] respectively.			
4.7.4.1.3C	The CS field shall be encoded as specified in Table 4-7.			
4.7.4.1.3D	The 64 cs_field bits shall be encoded in pairs as specified in Table 4-8.			
4.7.4.1.3E	The characters encoding the CS channel shall be transmitted in the order of the bits they encode beginning with the character encoding CS field bits [0,1] and ending with the character encoding bits [62-63].			
4.7.4.1.3F	A CS field whose bits [32-63] are not the bit wise complement of bits [0-31] respectively shall be determined to be corrupted.			
4.7.4.1.3G	A received CS field that is determined to be truncated and/or corrupted shall be ignored and discarded.			
4.7.4.1.3H	Any error detected in a truncated and/or corrupted CS field that is determined to be the result of a transmission error and not the result of truncation, such as an “invalid” or “illegal” character, shall be reported as an input error.			
4.7.4.1.4	IDLE2 CS Field Use	Part 6, Sec. 4.7.4.1.4		

4.7.4.1.4A	A receiver may issue the following commands. Only one of these commands may be issued at a time. Reset emphasis; preset emphasis; modify the emphasis provided by tap(-1), if tap(-1) is implemented; and modify the emphasis of tap(+1), if tap(+1) is implemented			
4.7.4.1.4B	Specific command bits may be changed only when the ACK and NACK bits are both de-asserted and the CMD bit is either de-asserted or transitioning from de-asserted to asserted.			
4.7.4.1.4C	Once the CMD bit is asserted, the connected port will either assert ACK after accepting and executing the command or assert NACK if the command cannot be executed.			
4.7.4.1.4D	The assertion of ACK or NACK shall occur no more than 250usec after the assertion of CMD.			
4.7.4.1.4E	ACK and NACK shall never be asserted at the same time.			
4.7.4.1.4F	Once ACK or NACK is asserted in a CS field received by the port issuing the command, the CMD bit is de-asserted.			
4.7.4.1.4G	ACK or NACK, whichever is asserted, shall be de-asserted within 250usec of receipt of a CS field with the CMD bit deasserted.			
4.7.4.1.4H	If, for any reason, the connected port fails to assert ACK or NACK within 250usec of the assertion of CMD, CMD may be deasserted.			
4.7.4.1.4I	Once deasserted, CMD shall remain deasserted for at least 250usec before being reasserted.			
4.7.4.1.4J	A CS field command to increase the emphasis of tap(n) by one step shall cause the tap(n) coefficient to be made more negative by one step.			
4.7.4.1.4K	A command to decrease the emphasis of tap(n) by one step shall cause the tap(n) coefficient to be made more positive by one step.			
4.7.5	Idle Sequence Selection	Part 6, Sec. 4.7.5		
4.7.5A	LP-Serial links operating at greater than 5.5 GBaud per lane shall always use the IDLE2 sequence.			
4.7.5B	LP-Serial links operating a less than 5.5 GBaud per lane shall support use of the IDLE1 sequence and may support use of the IDLE2 sequence.			
4.7.5C	If a LP-Serial port is operating at less than 5.5 GBaud per lane, supports the IDLE2 sequence and its configuration allows it to use the IDLE2 sequence, the port shall transmit the IDLE2 sequence when it enters the SEEK state of the port initialization process. (The port initialization process is specified in Section 4.12.) Otherwise, a LP-Serial port operating at less than 5.5 GBaud per lane shall transmit the IDLE1 sequence when entering the SEEK state and shall use the IDLE1 on the link until the port reenters the SEEK state.			
4.7.5D	A LP-Serial port transmitting the IDLE2 sequence shall monitor the idle sequence it is receiving from the connected port. The port shall determine the idle sequence being received from the connected port using a lane for which lane_sync is asserted.			

4.7.5E	If the LP-Serial port that is transmitting the IDLE2 sequence receives IDLE2 from the connected port, IDLE2 shall be the idle sequence used on the link until the port reenters the SEEK state.			
4.7.5F	If the port receives IDLE1 from the connected port, the port shall switch to transmitting IDLE1 and IDLE1 shall be the idle sequence used on the link until the port reenters the SEEK state.			
4.8	Scrambling	Part 6, Sec. 4.8	This is used only for link operating with IDLE Sequence 2(IDLE2)	Inter-op
4.8.1	Scrambling Rules	Part 6, Sec. 4.8.1		
4.8.1A	If the idle sequence selection process specified in Section 4.7.5 has selected idle sequence 1 (IDLE1) for use on the link, no characters shall be scrambled before transmission on the link.			Inter-op
4.8.1B	If the idle sequence selection process has selected idle sequence 2 (IDLE2), control symbol and packet data characters shall be scrambled by the transmitter before transmission on the link and descrambled in the receiver upon reception.			Inter-op
4.8.1C	Special characters, CS field marker data characters, and CS field data characters shall not be scrambled before transmission.			
4.8.1D	Scrambling and descrambling of control symbol and packet data characters shall not be disabled for normal link operation.			Inter-op
4.8.1E	Setting the Data scrambling disable bit does not disable the use of the lane scramblers for the generation of pseudo-random data characters for the IDLE2 random data field. (See Section 6.6.10, "Port n Control 2 CSRs (Block Offset 0x54, 74, ... , 234)").			
4.8.1F	Scrambling and descrambling shall be done at the lane level.			
4.8.1G	Nx ports shall have a transmit scrambling and receive descrambling function for each of the N lanes.			
4.8.1H	In the transmitter, scrambling shall occur before 8B/10B encoding, and if the port is operating in Nx mode, after lane striping.			
4.8.1I	In the receiver, descrambling shall occur after 8B/10B decoding, and if the port is operating in Nx mode, before lane destriping.			
4.8.1J	The polynomial x17+x8+1 shall be used to generate the pseudo-random sequences that are used for scrambling and descrambling.			
4.8.1K	The bit serial output of the pseudo-random sequence generator shall be taken from the output of the register holding x17.			
4.8.1L	Control symbol and packet data characters shall be scrambled and descrambled by XORing the bits of each character with the output of the pseudo-random sequence generator.			
4.8.1M	The bits of each data character are scrambled/descrambled in order of decreasing significance. The most significant bit (bit 0) is scrambled/descrambled first, the least significant bit (bit 7) is scrambled/descrambled last.			

4.8.1N	The transmitter and receiver scrambling sequence generators shall step during all characters except R special characters.			
4.8.1O	To minimize any correlation between lanes when a port is transmitting on multiple lanes, the scrambling sequence applied to a given output lane of the port shall be offset from the scrambling sequence applied to any other output lane of the port by at least 64 bits.			
4.8.2	Descrambler Synchronization	Part 6, Sec. 4.8.2		
4.8.2A	Each lane descrambler shall synchronize itself to the scrambled data stream it is receiving by using the scrambling sequence extracted from the pseudo-random data characters received by the lane to re-initialize the state of the descrambler.			
4.8.2B	After a lane descrambler has been re-initialized, the next two descrambler sync tests, which are defined in Section 4.8.3, shall be used to verify descrambler synchronization. If the result of both lane descrambler sync tests is "pass", the descrambler shall be determined to be "in sync". Otherwise, the lane descrambler shall be determined to be "out of sync" and the resynchronization process shall be repeated.			
4.8.2C	A LP-Serial port that is operating with IDLE2 shall transmit a SYNC sequence (described below) before transmitting any link-request control symbol.			
4.8.2C1	The SYNC sequence shall be transmitted in parallel on each of the N active lanes of a link operating in Nx mode and shall immediately precede the link-request control symbol.			
4.8.2C2	If the link is operating in 1x mode, the last character of the SYNC sequence is immediately followed by the first character of the link-request.			
4.8.2C3	If the link is operating in Nx mode, the last column of the SYNC sequence is immediately followed by the column containing the first characters of the link-request.			
4.8.2C4	The SYNC sequence shall be comprised of four contiguous repetitions of a five character sequence that begins with a M special character immediately followed by 4 pseudo-random data characters, i.e. the SYNC sequence is MDDDD MDDDD MDDDD MDDDD.			
4.8.2C5	The pseudo-random data characters shall be generated in the same way as the pseudo-random data characters in the random data field of the IDLE2 idle frame are generated.			
4.8.2C6	The SYNC sequence will appear as four repetitions of ||M||D||D||D||D|| on a link operating in Nx mode.			
4.8.3	Descrambler Synchronization Verification	Part 6, Sec. 4.8.3		

4.8.3A	Each active lane of a LP-Serial port that is descrambling received control symbol and packet data characters shall, with the one exception stated below, perform a descrambler synchronization state check (descrambler sync check) whenever a descrambler sync check trigger event is detected in the received character stream of the lane.			
4.8.3B	A descrambler sync check trigger event is defined as the occurrence of one of the following character sequences in the received character stream of an active lane.			
4.8.3B1	A single K, M or R special character that is not part of a contiguous sequence of K, M and/or R special characters.			
4.8.3B2	A contiguous sequence of K and/or R special characters possibly followed by a M special character.			
4.8.3C	The descrambler sync check shall consist of inspecting the descrambled values of the four contiguous characters following the trigger sequence. These four characters are defined as the check field.			
4.8.3C1	The check field for the first type of trigger event shall be the four characters immediately following the K, M or R special character. See 3.B.1 above.			
4.8.3C2	The check field for the second type of trigger event that does not end with a M special character shall be the four characters immediately following the contiguous sequence of K and/or R special characters. See 3.B.2 above.			
4.8.3C3	The check field for the second type of trigger event that ends with a M special character shall be the four characters immediately following the M special character.			
4.8.3D	The exception to the rule stated above in 3.A that each descrambler sync check trigger sequence shall cause the receiving lane to execute a descrambler sync check is when the descrambler check trigger sequence begins in the four character check field of a previous trigger sequence. When this occurs, the trigger sequence shall not trigger a descrambler sync check.			
4.8.3E	If the descrambled value of each of the four characters in a check field is D0.0, the result of the descrambler sync test shall be “pass”. Otherwise, the result of the descrambler sync test shall be “fail” and the descrambler shall be determined to be “out of sync”.			
4.8.3F	If a descrambler sync test fails, the port shall immediately enter the Input Error-stopped state if it is not already in that state and resynchronize the descrambler.			Inter-op
4.8.3G	All control symbols and packet received while a lane descrambler is out of sync shall be ignored and discarded.			

4.8.3H	The cause field in the packet-not-accepted control symbol issued by the port on entering the Input Error-stopped state due to a sync check failure shall indicate "loss of descrambler sync".			Inter-op
4.9	1x Mode Transmission Rules	Part 6, Sec. 4.9		
4.9.1	1x Ports	Part 6, Sec. 4.9.1		
4.9.1A	A 1x LP-Serial port shall 8B/10B encode and transmit the character stream of delimited control symbols and packets received from the upper layers in the order the characters were received from the upper layers.			
4.9.1B	When neither control symbols nor packets are available from the upper layers for transmission, an idle sequence shall be fed to the input of the 8B/10B encoder for encoding and transmission.			
4.9.1C	On reception, the code-group stream is 8B/10B decoded and the resulting character stream of error free delimited control symbols and packets shall be passed to the upper layers in the order the characters were received from the link.			
4.9.1D	If the link is operating with idle sequence 2, control symbol and packet data characters shall be scrambled before transmission and descrambled after reception as specified in Section 4.8.		Required only when the link is operating with idle sequence 2 (IDLE2). See Part 6, Sec. 4.8 for more details.	
4.9.1E	Figure 4-9 shows the encoding and transmission order for a short control symbol transmitted over a LP-Serial link operating in 1x mode.			
4.9.1F	Figure 4-10 shows the encoding and transmission order for a packet transmitted over a 1x LP-Serial link.			
4.9.2	Nx Ports Operating in 1x Mode	Part 6, Sec. 4.9.2		
4.9.2A	When a Nx port is operating in 1x mode, the character stream of delimited control symbols and packets received from the upper layers shall be fed in parallel to both lanes 0 and R for encoding and transmission in the order the characters were received from the upper layers.			
4.9.2B	When neither delimited control symbols nor packets are available from the upper layers for transmission, an idle sequence shall be fed in parallel to both lane 0 and lane R for 8B/10B encoding and transmission on lanes 0 and R.			
4.9.2C	On reception, the code-group stream from either lane 0 or R shall be selected according to the state of the 1x/Nx_Initialization state machine (Section 4.12.4.5), decoded and the error free delimited control symbols and packets passed to the upper layers.			
4.9.2D	When a port that optionally supports and is enabled for both 2x mode and a wider Nx mode is operating in 1x, the port shall support both lanes 1 and 2 as redundancy lanes.			
4.9.2E	The port shall transmit the 1x mode data stream on lanes 0, 1 and 2 and attempt to receive 1x mode data stream on lanes 0, 1 and 2.			
4.9.2F	The port shall select between using the data received on lane 0 or the data received on the redundancy lane which may be either lane 1 or lane 2 depending on the connected port.			

4.9.2G	Unless forced to use the redundancy lane, the port shall use the data stream received on lane 0 if it is available.			
4.9.2H	The 1x/Nx_Initialization state machine specified in Section 4.12.4.6 shall be modified for a port supporting both 2x and a wider Nx mode to comply with the above requirements.			
4.9.2I	If the link is operating with idle sequence 2, control symbol and packet data characters shall be scrambled before transmission and descrambled after reception as specified in Section 4.8.		Required only when the link is operating with idle sequence 2 (IDLE2). See Part 6, Sec. 4.8 for more details.	
4.10	Nx Link Striping and Transmission Rules	Part 6, Sec. 4.10		
4.10A	A LP-Serial port operating in Nx mode shall stripe the character stream of delimited control symbols and packets received from the upper layers across the N active output lanes in the order the characters were received from the upper layers.			
4.10B	Each lane shall then 8B/10B encode and transmit the characters assigned to it.			
4.10C	When neither control symbols nor packets are available from the upper layers for transmission, an idle sequence shall be fed to each of the N lanes for 8B/10B encoding and transmission.			
4.10D	Packets and delimited control symbols shall be striped across the N active lanes beginning with lane 0.			
4.10E	The first character of each packet, or delimited control symbol, shall be placed in lane K, where K modulo 4 = 0.			
4.10F	The second character shall be placed in lane (K + 1), and the nth character shall be placed in lane (K + (n - 1)) which wraps around to lane 0 when (K + (n - 1)) modulo N = 0.			
4.10G	When there are not enough delimiting control symbols or packets to fill a column, all remaining characters in the column shall be filled (padded) with pseudo-random data characters.		Required only when the link width is greater than 4.	
4.10H	The first pseudo-random data pad character shall occur in a lane whose lane_number modulo 4 = 0.		Required only when the link width is greater than 4.	
4.10I	The number of pseudo-random data pad characters in a column shall be a positive integer multiple of 4.		Required only when the link width is greater than 4.	
4.10J	Padding characters shall not be inserted between packet delimiting control symbols and the packet(s) they delimit.		Required only when the link width is greater than 4.	
4.10K	After striping, each of the N streams of characters shall be independently 8B/10B encoded and transmitted.			
4.10L	On reception, each lane shall be 8B/10B decoded.			
4.10M	If the link is operating with idle sequence 2, control symbol and packet data characters shall be scrambled before transmission and descrambled after reception as specified in Section 4.8.		Required only when the link is operating with idle sequence 2 (IDLE2). See Part 6, Sec. 4.8 for more details.	
4.10N	After decoding, the N lanes shall be aligned.			
4.10O	After alignment, the columns are destriped into a single character stream and passed to the upper layers.			

4.11	Retimers and Repeaters	Part 6, Sec. 4.11	Only required for retimers or repeaters	
4.11.1	Retimers	Part 6, Sec. 4.11.1	Only required for retimers	
4.11.1A	A retimer shall comply with all applicable AC specifications found in Part 6, Chapters 8-10.			
4.11.1B	Up to two retimers are allowed between 2 end nodes.			Inter-op
4.11.1C	A retimers may insert up to one /R/ code-group immediately following a /K/ code-group sequence, or remove one /R/ code-group that immediately follows a /K/ code-group sequence.			Inter-op
4.11.1D	An N-lane retimer must perform lane synchronization and deskew, in exactly the same way a RapidIO device implementing this physical layer does when synchronizing inputs during initialization and startup.			
4.11.1E	A Nx mode retimer will synchronize and align all lanes that are driven to it.			Inter-op
4.11.1F	If any link of a Nx mode retimer drops out, the retimer must merely continue to pass the active links.			Inter-op
4.11.1G	Any insertion or removal of a /R/ code-group in a N-lane retimer must be done on a full column.			Inter-op
4.11.1H	A retimers may retime links operating at the same width only (i.e. cannot connect a link operating at 1x to a link operating at 4x).			Inter-op
4.11.1I	Retimers do not check for code violations			Inter-op
4.11.2	Repeaters	Part 6, Sec. 4.11.2	Only required for repeaters	
4.11.2A	Repeaters do not compensate for clock rate variation.			
4.11.2B	Repeaters do not interpret or alter the bit stream in any way.			
4.12	Port Initialization	Part 6, Sec. 4.12		
4.12.1	1x Mode Initialization	Part 6, Sec. 4.12.1		
4.12.1A	The initialization process for ports that support only 1x mode shall be controlled by two state machines, 1x_Initialization and Lane_Synchronization.			
4.12.2	1x/Nx Mode Initialization	Part 6, Sec. 4.12.2		
4.12.2A	The initialization process for ports that support both 1x and a Nx mode is controlled by a primary state machine and four or more secondary state machines.			
4.12.2B	The primary state machine is the 1x/Nx_Initialization state machine.			
4.12.2C	Lane_Synchronization[0] through Lane_Synchronization[N-1] (one for each of the N lanes). Lane_Alignment (one for each supported Nx mode) and 1x/2x_Mode_Detect are the secondary state machines.		1x/2x_Mode_Detect is only used with the 1x/Nx initialization state machine	
4.12.3	Baud Rate Discovery	Part 6, Sec. 4.12.3	This functionality is optional.	
4.12.3A	Baud rate discovery occurs during the SEEK state of the 1x_Initialization and 1x/Nx_Initialization state machines.			
4.12.3B	Ports that implement baud rate discovery shall use the following algorithm.			

4.12.3B1	When the port enters the SEEK state, it begins transmitting an idle sequence on lane 0 and, if the port supports a Nx mode, on lane R, the 1x mode redundancy lane. The idle sequence shall be transmitted at the highest lane baud rate that is supported by the port and that is enabled for use.			
4.12.3B2	The port shall then look for an inbound signal on lane 0 or lane R of the link from a connected port.			
4.12.3B3	Once an inbound signal is detected, the port shall determine the baud rate of the signal.			
4.12.3B4	If the baud rate of the inbound signal is the same as the baud rate at which the port is transmitting, the link shall operate at that per lane baud rate until the port reenters the SEEK state and the baud rate discovery process is complete.			
4.12.3B5	If the baud rate on the inbound signal is less than the baud rate of the idle sequence transmitted by the port, the port shall reduce the baud rate at which it is transmitting to the next lowest baud rate that it supports and that is enabled for use and go to step 2.			
4.12.3B6	If the baud rate on the inbound signal is greater than the baud rate of the idle sequence being transmitted by the port, the port shall continue transmitting at the current baud rate and go to step 2.			
4.12.4	State Machines	Part 6, Sec. 4.12.4		
4.12.4.1.3	State Machine Variables	Part 6, Sec. 4.12.4.1.3		
4.12.4.1.3A	DCounter: DCounter behaves as specified in Table 4-10. The counter is used in the 1x/2x_Mode_Detect state machine			
4.12.4.1.3B1	disc_tmr_done (discovery timer done): Asserted when disc_tmr_en has been continuously asserted for 28 +/- 4 msec and the state machine is in the DISCOVERY or a RECOVERY state.			
4.12.4.1.3B2	disc_tmr_done (discovery timer done): The assertion of disc_tmr_done causes disc_tmr_en to be de-asserted.			
4.12.4.1.3B3	disc_tmr_done (discovery timer done): When the state machine is in a state other than the DISCOVERY or a RECOVERY state, disc_tmr done is de-asserted.			
4.12.4.1.3C	disc_tmr_en (discovery timer enable): When asserted, the discovery timer (disc_tmr) runs. When de-asserted, the discovery timer is reset to and maintains its initial value.			
4.12.4.1.3D	force_1x_mode: Asserted when all Nx (multi-lane) modes are disabled. When asserted, forces the 1x/Nx_Initialization state machine to use 1x mode.			

4.12.4.1.3E1	force_laneR: When force_1x_mode is asserted, force_laneR controls whether lane 0 or lane R, the redundancy lane , is preferred for 1x mode reception. If force_laneR is asserted, lane R is the preferred lane. If force_laneR is deasserted, lane 0 is the preferred lane. If the preferred lane is functional, it is selected by the port initialization state machine for 1x mode reception. If the preferred lane is not functional, the non-preferred lane, if functional, is selected for 1x mode reception.			
4.12.4.1.3E2	force_laneR: If force_1x_mode is not asserted, the state of force_laneR has no effect on the initialization state machine.			
4.12.4.1.3F	force_reinit: When asserted, forces the port Initialization state machine to re-initialize. The signal is set under software control and is cleared by the Initialization state machine.			
4.12.4.1.3G	Icounter: Counter used in the Lane_Synchronization state machine to count INVALID received code-groups. There is one Icounter for each lane in a Nx mode receiver.			
4.12.4.1.3H	Kcounter: Counter used in the Lane_Synchronization state machine to count received code-groups that contain a comma pattern. There is one Kcounter for each lane in a Nx mode receiver.			
4.12.4.1.3I	lanes01_drvr_oe: When asserted, the output drivers for lanes 0 and 1 are enabled.			
4.12.4.1.3J	lanes02_drvr_oe: When asserted, the output drivers for lanes 0 and 2 are enabled.			
4.12.4.1.3K	lanes13_drvr_oe: When asserted, the output drivers for lanes 1 and 3 are enabled.			
4.12.4.1.3L	link_drvr_oe: When asserted, the output link driver of a 1x port is enabled.			
4.12.4.1.3M	receive_lane1: In a 2x port that is initialized and is operating in 1x mode (2x_mode de-asserted), receive_lane1 indicates which lane the port has selected for input. When asserted, the port input is taken from lane 1. When de-asserted the port input is taken from lane 0. When the port is operating in 2x mode (2x_mode asserted), receive_lane1 is undefined and shall be ignored.			
4.12.4.1.3N	receive_lane2: In a Nx port that is initialized and is operating in 1x mode (Nx_mode de-asserted), receive_lane2 indicates which lane the port has selected for input. When asserted, the port input is taken from lane 2. When de-asserted the port input is taken from lane 0. When the port is operating in Nx mode (some Nx_mode asserted), receive_lane2 is undefined and shall be ignored.			
4.12.4.1.3O1	silence_timer_done: Asserted when silence_timer_en has been continuously asserted for 120 +/- 40 ms and the state machine is in the SILENT state. When the state machine is not in the SILENT state, silence_timer_done is de-asserted.			
4.12.4.1.3O2	silence_timer_done: The assertion of silence_timer_done causes silence_timer_en to be de-asserted.			

4.12.4.1.3P	silence_timer_en: When asserted, the silence_timer runs. When de-asserted, the silence_timer is reset to and maintains its initial value.			
4.12.4.1.3Q	Vcounter: Vcounter is used in the Lane_Synchronization state machine to count VALID received code-groups. There is one Vcounter for each lane in a Nx mode receiver.			
4.12.4.2	Lane Synchronization State Machine	Part 6, Sec. 4.12.4.2		
4.12.4.2A	A port that supports only 1x mode (1x port) has one Lane_Synchronization state machine.			
4.12.4.2B	A port that supports Nx mode has N Lane_Synchronization state machines, one for each lane (Lane_Synchronization[0] through Lane_Synchronization[N-1]).			
4.12.4.2C	The Lane_Synchronization state machine functions as specified in Figure 4-14.			
4.12.4.2D	An isolated single-bit or burst error shall not cause the code-group boundary alignment mechanism to change alignment.			
4.12.4.2E	The state machine starts in the NO_SYNC state and sets the variables Kcounter[n], Vcounter[n], and lane_sync[n] to 0. In this state it looks for a /COMMA/ code-group. When it finds one and signal_detect[n] is asserted, the state machine transitions to the NO_SYNC_1 state.			
4.12.4.2F	The NO_SYNC_1, NO_SYNC_2, and NO_SYNC_3 states look for the reception of 127 /COMMA/ and Vmin /VALID/ code-groups without any intervening /INVALID/ code-groups. When this condition is achieved, the state machine transitions to the SYNC state. If an intervening /INVALID/ code-group is detected, the state machine transitions back to the NO_SYNC state.			
4.12.4.2G	Vmin shall have a minimum value of 0.			
4.12.4.2H	When Vmin = 0, the behavior of this Lane_Synchronization state machine specified in Rev. 1.3 of this specification.			
4.12.4.2H	In the SYNC state, the state machine sets the variables lane_sync[n] to 1 and Icounter[n] to 0. In this state it looks for /INVALID/ code-groups. If an /INVALID/ code-group is detected, the state machine transitions to the SYNC_1 state.			
4.12.4.2I	The SYNC_1, SYNC_2, SYNC_3, and SYNC_4 states look for 255 consecutive /VALID/ code-groups without any /INVALID/ code-groups. When this condition is achieved, the Icounter[n] is decremented in the transition through the SYNC_4 state. If an /INVALID/ code-group is detected, the state machine increments Icounter[n]. If Icounter[n] decrements to 0, the state machine transitions back to the SYNC state. If Icounter[n] increments to Imax, the state machine transitions back to the NO_SYNC state.			
4.12.4.2J	Imax is an integer and shall have a value of 3 or greater for receivers not using DFE (decision feedback equalization) and a value of 4 or greater for receivers using DFE.			
4.12.4.3	Lane Alignment State Machine	Part 6, Sec. 4.12.4.3		

4.12.4.3A	A port supporting one or more multi-lane modes has one Lane_Alignment state machine for each supported Nx mode.			
4.12.4.3B	A port supporting only 1x mode does not have a Lane_Alignment state machine.			
4.12.4.3C	The Lane_Alignment state machine functions as specified in Figure 4-15.			
4.12.4.3D	Isolated single bit or burst errors shall not cause the lane alignment mechanism to change lane alignment.			
4.12.4.3E	The state machine starts in the NOT_ALIGNED state and sets the variables Acounter and N_lanes_aligned to 0. In this state it waits for all N lanes to achieve code-group boundary alignment and the reception of an ||A||. When this occurs, the state machine transitions to the NOT_ALIGNED_1 state.			
4.12.4.3F	The NOT_ALIGNED_1 and NOT_ALIGNED_2 states look for the reception of 4 ||A||s without the intervening reception of a misaligned column. When this occurs, the state machine transitions to the ALIGNED state. If an intervening misaligned column is received, the state machine transitions back to the NOT_ALIGNED state.			
4.12.4.3G	In the ALIGNED state, the state machine sets the variables N_lanes_aligned to 1 and Mcounter to 0 and looks for a misaligned column. If a misaligned column is detected, the state machine transitions to the ALIGNED_1 state.			
4.12.4.3H	The ALIGNED_1, ALIGNED_2, and ALIGNED_3 states look for the reception of 4 ||A||s without the intervening reception of more than Mmax - 1 additional misaligned columns. If this condition occurs, the state machine transitions back to the ALIGNED state. If Mmax - 1 additional intervening misaligned columns occurs, the state machine transitions back to the NOT_ALIGNED state.			
4.12.4.3I	Mmax is an integer and shall have a value of 2 or greater for receivers not using DFE and a value of 3 or greater for receivers using DFE.			
4.12.4.4	1x/2x Mode Detect State Machine	Part 6, Sec. 4.12.4.4		
4.12.4.4A	A port that supports 2x mode shall have one 1x/2x_Mode_Detect state machine.			
4.12.4.4B	The 1x/2x_Mode_Detect state machine functions as specified in Figure 4-16.			
4.12.4.4C	The 1x/2x_Mode_Detect state machine enters the INITIALIZE state whenever the port is reset or the state of 2_lanes_aligned changes state. The state machine initializes the variables 1x_mode_detected and Dcounter and waits for the lanes to become aligned. Once this occurs, the state machine transitions to the GET_COLUMN state.			

4.12.4.4D	In the GET_COLUMN state, each column is examined as it becomes available to determine whether it contains any control symbol delimiter special characters (SC or PD characters). If the column contains a single SC or PD special character, the state machine transitions to the 2X_DELIMITER state. If the column contains two SC or two PD special characters, the state machine transitions to the 1X_DELIMITER state.			
4.12.4.4E	In the 1X_DELIMITER state, Dcounter is decremented by 1 and its value is tested. If Dcounter is greater than 0, the state machine transitions back to the GET_COLUMN state. If Dcounter is 0, the state machine transitions to the SET_1X_MODE state, sets the variable 1x_mode_detected to 1, and then transitions back to the GET_COLUMN state.			
4.12.4.4F	In the 2X_DELIMITER state, Dcounter is incremented by 1 and its value is tested. If Dcounter is less than 3, the state machine transitions back to the GET_COLUMN state. If Dcounter is 3, the state machine transitions to the SET_2X_MODE state, sets the variable 1x_mode_detected to 0, and then transitions back to the GET_COLUMN state.			
4.12.4.5	1x Mode Initialization State Machine	Part 6, Sec. 4.12.4.5		
4.12.4.5A	The 1x_Initialization state machine shall be used by ports that support only 1x mode (1x ports).			
4.12.4.5B	The 1x_Initialization state machine functions as specified in Figure 4-17.			
4.12.4.5C	The state machine starts in the SILENT state and disables the link output driver. When the silence_timer expires, the state machine transitions to the SEEK state.			
4.12.4.5D	In the SEEK state, the link output driver is enabled, and an idle sequence is transmitted. When lane_ready and idle_selected are both asserted, the state machine transitions to the 1X_MODE state.			
4.12.4.5E	The input signal force_reinit allows the port to force link re-initialization at any time.			
4.12.4.5F	The variable port_initialized is asserted only in the 1X_MODE state.			
4.12.4.6	1x/Nx Mode Initialization State Machine for N = 4, 8, or 16	Part 6, Sec. 4.12.4.6		
4.12.4.6A	The 1x/Nx_Initialization state machine shall be used by ports that support both 1x mode and an Nx mode (1x/Nx ports) for N = 4, 8, or 16.			
4.12.4.6B	1x/8x and 1x/16x ports shall use the 1x/Nx_Initialization state machine specified in Figure 4-18.			
4.12.4.6C	1x/4x ports should use the 1x/Nx_Initialization state machine specified in Figure 4-18. The 1x/4x_Initialization state machine of Figure 4-19 shall not be used in new designs.			

4.12.4.6D1	The 1x/Nx_Initialization state machine starts in SILENT state. All N lane output drivers are disabled to force the link partner to re-initialize regardless of its current state. When the silent interval is complete, the state machine enters the SEEK state.			
4.12.4.6D2	The duration of the SILENT state is controlled by the silence_timer. The duration must be long enough to ensure that the link partner detects the silence (as a loss of lane_sync) and is forced to re-initialize.			
4.12.4.6E1	In the SEEK state, a 1x/Nx port transmits an idle sequence on lanes 0 and 2 (the other output drivers remain disabled to save power) and waits for an indication that a link partner is present.			
4.12.4.6E2	While lane_sync as defined indicates the bit and code-group boundary alignment state of a lane receiver, it is used by the state machine to indicate the presence of a link partner. A link partner is declared to be present when either lane_sync[0] or lane_sync[2] is asserted. The assertion of idle_selected and either lane_sync[0] or lane_sync[2] causes the state machine to enter the DISCOVERY state.			
4.12.4.6F1	In the DISCOVERY state, the port enables the output drivers for all N lanes and transmits an idle sequence on all N lanes if Nx mode is enabled. The discovery timer (disc_tmr) is started.			
4.12.4.6F2	While waiting for the end of the discovery period (disc_tmr_en asserted but disc_tmr_done de-asserted), if Nx mode is enabled, all N lanes become ready and lane alignment is achieved (N_lanes_ready asserted), the machine enters the Nx_MODE state.			
4.12.4.6F3	If force_1x_mode is asserted (Nx_mode_enabled is de-asserted), force laneR is not asserted and lane 0 becomes ready (lane_ready[0] asserted), the state machine transitions to the 1x_MODE_LANE0 state.			
4.12.4.6F4	If both force_1x_mode and force_laneR are asserted and lane 2 becomes ready (lane_ready[2] asserted), the state machine enters the 1x_MODE_LANE2 state.			
4.12.4.6F5	At the end of the discovery period (disc_tmr_done asserted), if the state machine has not entered the Nx_mode or one of the 1x modes and lane 0 is ready and either force_1x_mode and force_laneR are asserted but lane 2 is not ready or Nx mode is enabled but N_lanes_ready is de-asserted, the state machine will transition to the 1x_MODE_LANE0 state.			
4.12.4.6F6	At the end of the discovery period (disc_tmr_done asserted), if the state machine has not entered the Nx_mode or one of the 1x modes and lane 2 is ready, lane 0 is not ready and either force_1x_mode is asserted and force_laneR is not asserted or neither force_1x_mode nor N_lanes_ready are asserted, the state machine will transition to the 1x_MODE_LANE2 state.			

4.12.4.6F7	At the end of the discovery period (disc_tmr_done asserted), if the state machine has not entered the Nx_mode or one of the 1x modes and neither lane_ready[0] nor lane_ready[2] is asserted, the state machine will transition to the SILENT state and restart the port initialization process.			
4.12.4.6F8	If lane synchronization for both lane 0 and lane R is lost (both lane_sync[0] and lane_sync[2] de-asserted) during the DISCOVERY state, the state machine enters the SILENT state and restart the port initialization process.			
4.12.4.6G1	When in the Nx_MODE state, port_initialized is asserted.			
4.12.4.6G2	If N_lanes_ready is de-asserted, the state machine will transition to either the SILENT state if both lane_sync[0] and lane_sync[2] are de-asserted or the DISCOVERY state if either lane_sync[0] or lane_sync[2] is asserted.			
4.12.4.6H1	When in the 1x_MODE_LANE0 state, port_initialized is asserted.			
4.12.4.6H2	If lane_ready[0] is de-asserted but lane_sync[0] is still asserted, the state machine will transition to the 1x_RECOVERY state.			
4.12.4.6H3	If lane_sync[0] is de-asserted the state machine enters the SILENT state.			
4.12.4.6I1	When in the 1x_MODE_LANE2 state, port_initialized is asserted.			
4.12.4.6I2	If lane_ready[2] is de-asserted but lane_sync[2] is still asserted, the state machine will transition to the 1x_RECOVERY state.			
4.12.4.6I3	If lane_sync[2] is de-asserted the state machine enters the SILENT state.			
4.12.4.6J1	When the 1x_RECOVERY state is entered, the discovery timer (disc_tmr_en asserted) is started.			
4.12.4.6J2	The port transitions back to the 1x_MODE_LANE0 state if lane_ready[0] is re-asserted and the port was in the 1x_MODE_LANE0 state immediately before entering this state.			
4.12.4.6J3	The port transitions back to the 1x_MODE_LANE2 state if lane_ready[2] is re-asserted and the port was in the 1x_MODE_LANE2 state immediately before entering this state.			
4.12.4.6J4	If both lane_sync[0] and lane_sync[2] are lost (both lane_sync[0] and lane_sync[2] de-asserted), the state machine will transition to the SILENT state.			
4.12.4.6J5	If the appropriate lane_ready[] is not asserted before the discovery timer is up (disc_tmr_done asserted), the state machine will transition to the SILENT state.			
4.12.4.6K	The input signals force_1x_mode and force_laneR allow the state machine to be forced during initialization into 1x mode, and in 1x mode to be forced to receive on lane 2.			
4.12.4.6L	The input signal force_reinit allows the port to force port n link re-initialization at any time.			
4.12.4.6M	The variable port_initialized is asserted only in the 1X_MODE_LANE0, 1x_MODE_LANE2, and Nx_MODE states.			

4.12.4.7	1x/2x Mode Initialization State Machine	Part 6, Sec. 4.12.4.7		
4.12.4.7A	The 1x/2x_Initialization state machine specified in Figure 4-20 shall be used by 1x/2x ports.			
4.12.4.7B	Except for the method it uses to decide whether to operate in 1x or 2x mode and the use of lane 1 as the redundancy lane, this state machine is identical to the 1x/Nx_Initialization state machine specified in Figure 4-18 with N = 2.			
4.12.4.7C	Ports that support more than 2 lanes disable all lanes except lanes 0 and R when operating in 1x mode.			
4.12.4.7D	1x/2x ports transmit on both lanes 0 and R regardless of whether they are operating in 1x or 2x mode.			
4.12.4.7E	The 1x/2x_Mode_Detect state machine specified in Section 4.12.4.4 provides the mechanism to determine whether to operate in 1x or 2x mode.			
4.12.4.8	1x/Mx/Nx Mode Initialization State Machines	Part 6, Sec. 4.12.4.8		
4.12.4.8A1	The negotiation algorithm implemented by the state machine attempts to select the greatest link width supported by both ports of a connected port pair.			
4.12.4.8A2	Once a link width is selected, a wider link width can be selected only if the state machine enters the SILENT state which restarts the selection algorithm.			
4.12.4.8.1	1x/2x/Nx Initialization State Machine	Part 6, Sec. 4.12.4.8.1		
4.12.4.8.1A	The 1x/2x/Nx_Initialization state machine is specified in Figure 4-21 and shall be used by 1x/2x/Nx ports.			
4.12.4.8.1B	The 1x/2x/Nx_Initialization state machine has three more states than a 1x/Nx_Initialization state machine: the 2x_MODE, 2x_RECOVERY, and 1x_MODE_LANE1 states.			
4.12.4.8.1C	The operation of the 1x/2x/Nx_Initialization state machine is essentially the same as that of a 1x/2x_Initialization state machine for the 1x and 2x modes operation and that of a 1x/Nx_Initialization state machine for the Nx mode operation. The differences between the 1x/2x/Nx_Initialization state machine and the others (1x/2x_Initialization and 1x/Nx_Initialization) are as follows.			
4.12.4.8.1C1	In the SEEK state, the lanes whose drivers are output enabled depend on the modes that are enabled. Lanes 0 and 1 are output enabled if the 2x mode is enabled. Lanes 0 and 2 are output enabled if the Nx mode is enabled or the 2x mode is disabled. And if both modes are enabled, lanes 0, 1, and 2 are output enabled.			
4.12.4.8.1C2	The state machine enters the DISCOVERY state when lane_sync is asserted for lanes 0, 1, or 2.			
4.12.4.8.1C3	In the DISCOVERY state, the lane selection priority for 1x mode is lane 0 first, lane 2 second, and lane 1 third.			
4.12.4.8.1C4	In the 2x_MODE state, the state machine transitions to the 2x_RECOVERY state if 1x_mode_detected is asserted.			

4.12.4.8.1C5	In the 2x_RECOVERY state, the state machine then transitions to the 1x_MODE_LANE0 state if both 2_lanes_ready and 1x_mode_detected are still asserted.			
4.12.4.8.2	1x/Mx/Nx Initialization State Machine (N > M > 2)			
4.12.4.8.2A	The 1x/Mx/Nx_Initialization state machine for N > M > 2 is specified in Figure 4-22 and shall be used by 1x/Mx/Nx ports.			
4.12.4.8.2B	The 1x/Mx/Nx_Initialization state machine has two more states than a 1x/Nx_Initialization state machine: the Mx_MODE and Mx_RECOVERY states, but one less state than the 1x/2x/Nx_Initialization state machine: the 1x_MODE_LANE1 state.			
4.12.4.8.2C	Its operation is most similar to that of the 1x/2x/Nx_Initialization state machine, but is less complex as the redundancy lane R is the same for all N and M > 2.			

Item #	Compliance Item	Specification Reference	Optional	Interop Item
5	LP-Serial Protocol	Part 6, Chap. 5		
5.4	Virtual Channels	Part 6, Sec. 5.4		
5.4.1	Virtual Channel 0 (VC0)	Part 6, Sec. 5.4.1		
5.4.1A	VC0 shall be supported by all LP-Serial ports.			Inter-op
5.4.1B	VC0 shall always be active, operate in RT mode and support packet priority rules.			Inter-op
5.4.2	Virtual Channels 1-8 (VC1-8)	Part 6, Sec. 5.4.2	Optional	
5.4.2A	Any of VC1 through VC8 that are implemented shall support operation in RT mode and may optionally support and be configured for operation in CT mode.		Required in multi-VC mode	Inter-op
5.4.2B	CT VCs operate independent of each other.			
5.4.2C	RT VCs operate as a “RT Group”. That is to say, when the error recovery protocol is used to recover a damaged packet, the unacknowledged packets for all VCs in RT mode are retransmitted.			
5.4.2D	Implementations with fewer than the full number of VCs should ignore, but must not modify, any ignored VC bits.			Inter-op
5.4.2E	The number of channels for VCs 1-8 may be 0, 1, 2, 4, or 8. The hierarchy for combining VCs shall follow Part 6 Table 5-1.		Required in multi-VC mode	Inter-op
5.4.3	Virtual Channel Utilization	Part 6, Sec. 5.4.3	Required in multi-VC mode	
5.4.3A	Packets are transmitted from one or more virtual channels according to the weighted distribution of bandwidth for each channel. The weighting is such that under demand for full utilization of the link’s bandwidth, each active VC is guaranteed a certain portion of that bandwidth.			
5.4.3B	There are no packet ordering guarantees between VCs.			Inter-op
5.4.3C	Packets within a VC in VCs 1 - 8 are equally weighted and must be kept in order.			Inter-op
5.5	Control Symbols	Part 6, Sec. 5.5		
5.5A	Control symbol field and format shall be implemented according to Part 6, Chapter 3.			
5.5.1	Control Symbol Selection	Part 6, Sec. 5.5.1		
5.5.1A	The control symbol used on a LP-Serial link is determined by the idle sequence. being used on the link. Idle sequence selection occurs during the port initialization process			
5.5.1B	If the link is operating with idle sequence 1 (IDLE1), the short control symbol shall be used.			Inter-op
5.5.1C	If the link is operating with idle sequence 2 (IDLE2), the long control symbol shall be used.		Required for idle2	
5.5.2	Control Symbol Delimiting	Part 6, Sec. 5.5.2		
5.5.2A	Short control symbols are delimited by a single 8B/10B special character that marks the beginning of the control symbol and immediately precedes the first character of the control symbol.			Inter-op

5.5.2B	Long control symbols are delimited by two 8B/10B special characters. The first special character marks the beginning of the control symbol (the start delimiter) and immediately precedes the first character of the control symbol. The second special character marks the end of the control symbol (the end delimiter) and immediately follows the last character of the control symbol.			Inter-op
5.5.2C	The end delimiter special character replicates the value of the start delimiter special character.			
5.5.2D	One of two special characters is used to delimit a control symbol. If the control symbol contains a packet delimiter, the special character PD (K28.3) is used. If the control symbol does not contain a packet delimiter, the special character SC (K28.0) is used.			Inter-op
5.5.2E	The control symbol delimiting special character(s) shall be added to the control symbol before the control symbol is passed to the PCS sublayer for 8B/10B encoding and, if applicable, lane striping.			Inter-op
5.5.3	Control Symbol Use	Part 6, Sec. 5.5.3		
5.5.3.1	Link Initialization	Part 6, Sec. 5.5.3.1		
5.5.3.1A	An LP-Serial port that is not initialized only transmits an idle sequence.			
5.5.3.1B	When a port is in the port_initialized state, but not in the link_initialized state, the port shall transmit only a idle sequences, status, VC-status, link-request and link-response control symbols and, if IDLE2 is the idle sequence in use on the link, SYNC sequences.			Inter-op
5.5.3.1C	After a LP-Serial port is initialized, the port shall complete the following sequence of actions to enter the link_initialized state.			
5.5.3.1C1	The initialized port shall transmit idle and at least one status control symbol per 1024 code-groups transmitted per lane until the port has received an error free status control symbol from the connected port.			Inter-op
5.5.3.1C2	After the initialized port has received an error free status control symbol from the connected port, the port shall transmit idle and at least 15 additional status control symbols.			Inter-op
5.5.3.1C3	After the initialized port has received an error free status control symbol, the port shall wait until it has received a total of seven error free status control symbols with no intervening errors.			Inter-op
5.5.3.1C4	If any VC other than VC0 is implemented and enabled, the port shall transmit a single VC_Status control symbol for each such VC.			Inter-op
5.5.3.1D	Once a port is in the link_initialized state, loss of port initialization shall cause the port to exit the link_initialized state.			

5.5.3.1E	Once the port has exited the link_initialized state, the port shall not resume the normal transmission of packets and control symbols until the port has re-entered both the port_initialized and link_initialized states. -- to be checked on this statement, suggest change: Once the port has exited the link_initialized state, the port shall not resume the normal transmission of packets until the port has re-entered both the port_initialized and link_initialized states			
5.5.3.1F	A port that is not in the port_initialized state shall ignore and discard any packet or control symbol that it receives from the connected port.			Inter-op
5.5.3.1G	A port that is in the port_initialized state but not in the link_initialized state shall ignore and discard any packet or any control symbol, other than status, VC-status, link-request or link-response control symbols, that it receives from the connected port.			Inter-op
5.5.3.1H	A LP-Serial port shall not enter the Input error-stopped state or the Output error-stopped state unless the port is in the link_initialized state.			Inter-op
5.5.3.1I	The loss of link initialization shall not cause a port already in the Input error-stopped state or the Output error-stopped state to exit either of those states.			
5.5.3.2	Buffer Status Maintenance	Part 6, Sec. 5.5.3.2		
5.5.3.2A	When a LP-Serial port is in the normal operational state, it shall transmit a control symbol containing the buf_status field for VC0 at least once every 1024 code-groups transmitted per lane.			Inter-op
5.5.3.2B	When a LP-Serial port is in the normal operational state and any VC other than VC0 is active (VCs 1-8), the port shall transmit a control symbol containing the buf_status field for each active VC at least once every VC refresh period.			Inter-op
5.5.3.2C	The VC refresh period can be configured through the VC Refresh Interval register field defined in Chapter 6, "LP-Serial Registers".			
5.5.3.2D	The shortest VC refresh period is 1024 code-groups and the longest required VC refreshing period is 1024 x 16 = 16K code groups. The VC refresh period must be implemented supporting 16K code groups.			
5.5.3.2E	The VC Refresh Interval register field contains space for up to 8 bits to be used, so based on implementation, the maximum refresh period may be 256K code groups.		Optional	
5.5.3.3	Embeded Control Symbols	Part 6, Sec. 5.5.3.3		
5.5.3.3A	Any control symbol that does not contain a packet delimiter may be embedded in a packet.			
5.5.3.3B	When a control symbol is embedded in a packet, the delimited control symbol shall begin on a 4-character boundary of the packet.			Inter-op
5.5.3.4	Multicast-Event Control Symbols	Part 6, Sec 5.5.3.4		

5.5.3.4A	When a switch processing element receives a Multicast-Event control symbol, the switch shall forward the Multicast-Event by issuing a Multicast-Event control symbol from each port that is designated in the port's CSR as a Multicast-Event output port.			Inter-op
5.5.3.4B	A switch port shall never forward a Multicast-Event control symbol back to the device from which it received a Multicast-Event control symbol regardless of whether the port is designated a Multicast-Event output or not.			Inter-op
5.5.3.4C	In the event that two or more Multicast-Event control symbols are received by a switch processing element close enough in time that more than one is present in the switch at the same time, at least one of the Multicast-Event control symbols shall be forwarded. The others may be forwarded or discarded .			
5.5.3.4D	Multicast-Event control symbols have the highest priority for transmission on a link and can be embedded in packets.			
5.5.3.4E	The maximum value of Multicast-Event forwarding delay and delay variation shall be defined in switch device specification.		Required for switch	
5.6	Packets	Part 6, Sec. 5.6		
5.6.1	Packet Delimiting	Part 6, Sec. 5.6.1		
5.6.1A	LP-Serial packets are delimited for transmission by control symbols. Since packet length is variable, both start-of-packet and end-of-packet delimiters are required.			
5.6.1B	The start-of-packet delimiter immediately precedes the first character of the packet or an embedded delimited control symbol.			Inter-op
5.6.1C	The control symbol marking the end of a packet (packet termination) immediately follows the last character of the packet or the end of an embedded delimited control symbol.			Inter-op
5.6.1D	The following control symbols are used to delimit packets: Start-of-packet, End-ofpacket, Stomp, Restart-from-retry and any Link-request.			
5.6.1.1	Packet Start	Part 6, Sec. 5.6.1.1		
5.6.1.1A	The beginning of a packet shall be marked by a start-of-packet control symbol.			Inter-op
5.6.1.2	Packet Termination	Part 6, Sec. 5.6.1.2		
5.6.1.2A	A packet shall be terminated in one of the following three ways:			Inter-op
5.6.1.2A1	The end of a packet is marked with an end-of-packet control symbol.			
5.6.1.2A2	The end of a packet is marked with a start-of-packet control symbol that also marks the beginning of a new packet.			
5.6.1.2A3	The packet is canceled by a restart-from-retry, stomp, or any link-request control symbol.			
5.6.2	Acknowledgment Identifier	Part 6, Sec. 5.6.2		
5.6.2A	Each packet requires an identifier to uniquely identify its acknowledgment control symbol. The acknowledge ID (ackID) is 5 bits long when using short control symbols and 6 bits long when using long control symbols.		Long control symbol is required when IDLE2 is used	Inter-op

5.6.2B	A maximum of 2N-1 outstanding unacknowledged packets shall be allowed at any one time(N is the number of bits in the ackID field).			Inter-op
5.6.2C	The value of ackID assigned to the first packet transmitted after a reset shall be 0.			
5.6.2D	The values of ackID assigned to subsequent packets shall be in increasing numerical order, wrapping back to 0 on overflow.			Inter-op
5.6.2E	The ackID assigned to a packet indicates the order of the packet transmission and is independent of the virtual channel assignment of the packet. When acknowledgement control symbols are received containing VC specific information (e.g., buf_status), the transmitter side of the port must reassociate that information with the correct VC based on the returned ackID.		Required when multiple VCs are enabled	Inter-op
5.6.3	Packet Priority and Transaction Request Flows	Part 6, Sec. 5.6.3		
5.6.3A	Within VC0 each packet has a priority,The priority is carried in the prio field of the packet and has four possible values: 0, 1, 2, or 3.			Inter-op
5.6.3B	Packet priority increases with the priority value with 0 being the lowest priority and 3 being the highest.			Inter-op
5.6.3C	Within VC0 each packet has optionally a critical request flow. The critical request flow is carried in the CRF bit.		Optional	Inter-op
5.6.3D	Devices that do not support the CRF bit treat it as reserved, setting it to logic 0 on transmit and ignoring it on receive.			Inter-op
5.6.3E	Packets with the same priority level and CRF bit setting cannot pass each other. Packets with the CRF bit set at a given priority are allowed to pass packets with the CRF bit clear at the same priority.			Inter-op
5.6.3F	When a transaction is encapsulated in a packet for transmission, the transaction request flow indicator (flowID) of the transaction is mapped into the prio field (and optionally the CRF bit) of the packet.			Inter-op
5.6.3G	If the CRF bit is not supported, VC0 transaction request flows are mapped according to Part 6, Table 5-2.			Inter-op
5.6.3H	If the CRF bit is supported, the VC0 transaction request flows are mapped according to Part 6, Table 5-3.			Inter-op
5.6.3I	Flows for VCs 1-8 (A and higher) are mapped according to Part 6, Table 5-4.		Optional	Inter-op
5.6.3J	Transaction requests that require responses, and their corresponding responses, must use VC0 with the appropriate priority.			Inter-op
5.7	Link Maintenance Protocol	Part 6, Sec. 5.7		
5.7A	For software management, the request is generated through ports in the configuration space of the sending device. An external host write of a command to the link-request register with an I/O logical specification maintenance write transaction causes a link-request control symbol to be issued onto the output port of the device.			Inter-op

5.7B	Only one link-request can be outstanding on a link at a time.			
5.7C	The device that is linked to the sending device shall respond with a link-response control symbol if the link-request command required it to do so.			Inter-op
5.7D	The external host retrieves the link-response by polling the link-response register with I/O logical maintenance read transactions.			
5.7E	A device with multiple RapidIO interfaces has a link-request and a link-response register pair for each corresponding RapidIO interface.			Inter-op
5.7F	The automatic error recovery mechanism relies on the hardware generating link-request/input-status control symbols under the transmission error conditions described in Section 5.13.2.1, "Recoverable Errors" and using the corresponding link-response information to attempt recovery.			
5.7G	A device receiving a link-request/reset-device control symbol shall not perform the reset function unless it has received four link-request/reset-device control symbols in a row without any intervening packets or other control symbols, except status control symbols.			Inter-op
5.7H	The link-request/reset-device control symbol does not require a response.			
5.7I	The input-status command of the link-request/input-status control symbol is used by the hardware to recover from transmission errors.			
5.7J	If the input port had stopped due to a transmission error that generated a packet-not-accepted control symbol back to the sender, the link-request/input-status control symbol acts as a link-request/restart-from-error control symbol, and the receiver is re-enabled to receive new packets after generating the link-response control symbol.			Inter-op
5.7K	The link-request/input-status control symbol may also be used to restart the receiving device if it is waiting for a restart-from-retry control symbol after retrying a packet.			Inter-op
5.7L	The link-request/input-status control symbol requires a response.			Inter-op
5.7L1	A port receiving a link-request/input-status control symbol returns a link-response control symbol which contains the “port_status” and “ackID_status”.			Inter-op
5.7L2	The status indications are limited to the definitions which are described in Table 3-6.			
5.7M	The retry-stopped state indicates that the port has retried a packet and is waiting to be restarted. This state is cleared when a restart-from-retry (or a link-request/input-status) control symbol is received.			Inter-op
5.7N	The error-stopped state indicates that the port has encountered a transmission error and is waiting to be restarted. This state is cleared when a link-request/input-status control symbol is received.			Inter-op

5.8	Packet Transmission Protocol	Part 6, Sec. 5.8		
5.8A	Each packet transmitted across a LP-Serial link shall be acknowledged by the receiving port with a packet acknowledgment control symbol with one exception as stated in 5.8C.			
5.8B	Packets shall be acknowledged in the order in which they were transmitted (ackID order) with exception stated in 5.8C.			Inter-op
5.8C	The exception is when an event has occurred that caused a port to enter the Input Error-stopped state. CT mode packets accepted by a port after the port entered the Input Error-stopped state and before the port receives a link-request/input-status control symbol shall not be acknowledged.		Required when CT mode is enabled	Inter-op
5.8D	To associate packet acknowledgment control symbols with transmitted packets, each packet shall be assigned an ackID value according to the rules of Section 5.6.2, "Acknowledgment Identifier" that is carried in the ackID field of the packet and the packet_ackID field of the associated acknowledgment control symbol.			Inter-op
5.8E	The LP-Serial link RT protocol uses retransmission to recover from packet transmission errors or a lack of receive buffer resources. To enable packet retransmission, a copy of each RT packet transmitted across a LP-Serial link shall be kept by the sending port until either a packet-accepted control symbol is received for the packet or the sending port determines that the packet has encountered an unrecoverable error condition.			Inter-op
5.8F	The LP-Serial link CT protocol does not use packet retransmission. CT mode packets that are corrupted by transmission errors or that are not accepted because of a lack of receive buffer resources are discarded and lost. Therefore, a port need not retain a copy of a CT mode packet whose transmission has been completed.		Required when CT mode is enabled	Inter-op
5.8G	The LP-Serial link protocol uses the ackID value carried in each packet to ensure that no RT mode packets are lost due to transmission errors. A port shall accept packets from a LP-Serial link only in sequential ackID order with one exceptions as stated in 5.8H and 5.8I.			Inter-op
5.8H	The exception is when an event has occurred that caused a port to enter the Input Error-stopped state. A CT mode packet received by a port after the port entered the Input Error-stopped state and before the port receives a link-request/input-status control symbol shall be accepted by the port without regard to the value of the packet's ackID field if the packet is otherwise error free and there are adequate receive buffer resources to accept the packet.		Required when CT mode is enabled	Inter-op
5.8I	The value that is maintained by the port of the ackID expected in the next packet shall not be changed by the acceptance of CT packets after the port entered the Input Error-stopped state and before the port receives a link-request/input-status control symbol.		Required when CT mode is enabled	Inter-op

5.8J	In order to prevent switch processing element internal errors, such as SRAM soft bit errors, from silently corrupting a packet and the system,switch processing elements shall maintain packet error detection coverage while a packet is passing through the switch.		Required only for switches	
5.9	Flow Control	Part 6, Sec. 5.9		
5.9A	Every RapidIO LP-Serial port shall support receiver-controlled flow control as implemented according to Section 5.9.1			Inter-op
5.9.1	Receiver-Controlled Flow Control	Part 6, Sec. 5.9.1		
5.9.1A	A port signals its link partner that it is operating in receiver-controlled flow control mode by setting the buf_status field to all 1's in every control symbol containing the field that the port transmits.			Inter-op
5.9.1B	A port operating in receiver-controlled flow control mode accepts or rejects each inbound error free packet based on whether the receiving port has enough buffer space available for the VC and the priority level of the packet.			Inter-op
5.9.1.1	Reliable Traffic VC Receivers	Part 6, Sec. 5.9.1.1		
5.9.1.1A	If buffer space is not available, the port rejects the packet.			
5.9.1.1B	If multiple VCs are active, and the VC is in reliable traffic mode, the rejected packet shall be acknowledged with the packet-not-accepted control symbol.		Only applicable if multiple VCs are supported	Inter-op
5.9.1.2	Continuous Traffic VC Receivers	Part 6, Sec. 5.9.1.2		
5.9.1.2A	If buffer space is not available, and the VC is in CT mode, the packet is acknowledged as accepted, and the packet is discarded.			
5.9.1.3	Single VC Retry Protocol	Part 6, Sec. 5.9.1.3		
5.9.1.3A	It is a requirement that implementers include this functionality (retry prtotocol) in the channel design to be backward compatible with existing RapidIO interfaces.			Inter-op
5.9.1.3B	When a port rejects a packet, it immediately enters the Input Retry-stopped state and follows the Input Retry-stopped recovery process specified in Section 5.9.1.4, "Input Retry-Stopped Recovery Process".			Inter-op
5.9.1.3C	A port that receives a packet-retry control symbol immediately enters the Output Retry-stopped state and follows the Output Retry-stopped recovery process specified in Section 5.9.1.5, "Output Retry-Stopped Recovery Process".			Inter-op
5.9.1.4	Input Retry-Stopped Recovery Process	Part 6, Sec. 5.9.1.4		
5.9.1.4A	When the input side of a port operating with only VC0 active (single VC mode) retries a packet, it immediately enters the Input Retry-stopped state. To recover from this state, the input side of the port takes the following actions.			Inter-op
5.9.1.4B	Discards the rejected or canceled packet without reporting a packet error and ignores all subsequently received packets while the port is in the Input Retry-stopped state.			
5.9.1.4C	Causes the output side of the port to issue a packet-retry control symbol containing the ackID value of the retried packet in the packet_ackID field of the control symbol.			

5.9.1.4D	When a restart-from-retry control symbol is received, exit the Input Retry-stopped state and resume packet reception.			
5.9.1.5	Output Retry-Stopped Recovery Process	Part 6, Sec. 5.9.1.5		
5.9.1.5A	To recover from the Output Retry-stopped state, the output side of a port takes the following actions.			Inter-op
5.9.1.5B	Immediately stops transmitting new packets.			Inter-op
5.9.1.5C	Resets the link packet acknowledgment timers for all transmitted but unacknowledged packets.			
5.9.1.5D	Transmits a restart-from-retry control symbol.			Inter-op
5.9.1.5E	Backs up to the first unaccepted packet (the retried packet) which is the packet whose ackID value is specified by the packet_ackID value contained in the packet-retry control symbol.			Inter-op
5.9.1.5F	Exits the Output Retry-stopped state and resumes transmission with either the retried packet or a higher priority packet which is assigned the ackID value contained in the packet_ackID field of the packet-retry control symbol.			Inter-op
5.9.2	Transmitter-Controlled Flow Control	Part 6, Sec. 5.9.2	Only required if port is supporting Transmitter-Controlled Flow control	
5.9.2A	A port signals its link partner that it is operating in transmitter-controlled flow control mode by setting the buf_status field to a value different from all 1's in every control symbol containing the field that the port transmits.			Inter-op
5.9.2B	The value conveyed by the buf_status field is the number of maximum length packet buffers currently available for packet reception up to the limit that can be reported in the field.			
5.9.2C	A port informs its link partner when the number of free buffers available for packet reception changes.			
5.9.2D	A port shall send a control symbol containing the buf_status field to its link partner no less often than the minimum rate specified in Section 5.5.3.2, "Buffer Status Maintenance".			
5.9.2E	When a port implements more than VC0, the value of buf_status is kept on a per VC basis by the receiving port.		Only required if port is supporting Transmitter-Controlled Flow control and if multiple VCs are supported	
5.9.2F	When a packet-accepted symbol is returned, the buf_status field is filled with the status for the specific VC that the packet was sent to.		Only required if port is supporting Transmitter-Controlled Flow control and if multiple VCs are supported	
5.9.2G	When sending buf_status asynchronously (not in response to any specific packet), the status control symbol is used for VC0, and the VC_status control symbol is used for VC's 1-8.		Only required if port is supporting Transmitter-Controlled Flow control and if multiple VCs are supported	

5.9.2H	A port whose link partner is operating in transmitter-control flow control mode should never receive a packet-not-accepted (or packet-retry control symbol if operating in single VC mode) from its link partner unless the port has transmitted more packets than its link partner has receive buffers, has violated the rules that all input buffers may not be filled with low priority packets or there is some fault condition.			
5.9.2I	If a port, operating in single VC mode, for whose link partner is operating in transmitter-control flow control mode, receives a packet-retry control symbol, the output side of the port immediately enters the Output Retry-stopped state and follows the Output Retry-stopped recovery process specified in Section 5.9.1.5, "Output Retry-Stopped Recovery Process".			Inter-op
5.9.2.1	Receive Buffer Management	Part 6, Sec. 5.9.2.1		
5.9.2.1A	In transmitter-controlled flow control, the transmitter manages the packet receive buffers in the receiver and shall not violate the rules in Section 5.12, "Deadlock Avoidance" concerning the acceptance of packets by ports.			
5.9.2.1B	For VCs 1-8, packets within the same VC are equal in priority and always kept in order. The only requirement is that once a given amount of buffers is reported by the receiver to the transmitter those buffers shall remain available for packets for that VC.		Only applicable if multiple VCs are supported	
5.9.2.2	Effective Number of Free Receive Buffers	Part 6, Sec. 5.9.2.2		
5.9.2.2A	The value in the buf_status field does not account for packets that have been transmitted by the VC but not acknowledged by its link partner.			
5.9.2.3	Speculative Packet Transmission	Part 6, Sec. 5.9.2.3		
5.9.2.3A	The link partner accepts or rejects these (speculatively) packets on a packet by packet basis in exactly the same way it would if operating in receiver-controlled flow control mode.			
5.9.3	Flow Control Mode Negotiation	Part 6, Sec. 5.9.3		
5.9.3A	Immediately following the initialization of a link, each port begins sending status control symbols to its link partner. The value of the buf_status field in these control symbols indicates to the link partner the flow control mode supported by the sending port.			
5.9.3B	If the port and its link partner both support transmitter-controlled flow control, then both ports shall use transmitter-controlled flow control. Otherwise, both ports shall use receiver-controlled flow control.			
5.9.3C	If multiple VCs are used, then a port shall have either all channels in receiver based flow control or all channels in transmitter based flow control. All status and VC_status control symbols shall be consistent in their buf_status reporting in this regard.		Only applicable if multiple VCs are supported	
5.10	Canceling Packets	Part 6, Sec 5.10		

5.10A	The sending device shall use the stomp control symbol, the restart-from-retry control symbol (in response to a packet-retry control symbol), or any link request control symbol to cancel a packet.			Inter-op
5.10B	A port receiving a canceled packet shall drop the packet. The cancellation of a packet shall not result in the generation or report of any errors.			
5.10C	If the packet was canceled because the sender received a packet-not-accepted control symbol, the error that caused the packet-not-accepted to be sent shall be reported in the normal manner.			Inter-op
5.10D	A port that is not in an input stopped state (Retry-stopped or Error-stopped) while receiving the canceled packet and has not previously acknowledged the packet shall have the following behavior.			
5.10E	If the packet is canceled by a link-request/input-status control symbol, the port shall drop the packet without reporting a packet error.			
5.10F	If the packet is canceled by a restart-from-retry control symbol a protocol error has occurred and the port shall immediately enter the Input Error-stopped state and follows the Input Error-stopped recovery process specified in Section 5.13.2.6, "Input Error-Stopped Recovery Process".			Inter-op
5.10G	If the packet was canceled by other than a restart-from-retry or link-request/input-status control symbol and the port is operating in single VC mode (only VC0 is active), the port shall immediately enter the Input Retry-Stopped state and follow the Input Retry-Stopped recovery process specified in Section 5.9.1.4, "Input Retry-Stopped Recovery Process".			Inter-op
5.10H	If the packet was canceled before the packet ackID field was received by the port, the packet_ackID field of the associated packet-retry control symbol acknowledging the packet shall be set to the ackID the port expected in the canceled packet.			Inter-op
5.10I	If the packet was canceled by other than a restart-from-retry or link-request/input-status control symbol and the port is operating in multiple VC mode (at least one of VC1-8 is active), the port shall immediately enter the Input Error-Stopped state and follow the Input Error-Stopped recovery process specified in Section 5.13.2.6, "Input Error-Stopped Recovery Process".		Only applicable if multiple VCs are suppported	Inter-op
5.10J	A packet whose transmission is canceled shall be considered to be an untransmitted packet.			
5.11	Transaction and Packet Delivery Ordering Rules	Part 6, Sec. 5.11		
5.11A	The device is compliant to transaction delivery ordering rules			
5.11A1	The physical layer of an end point processing element port shall encapsulate in packets and forward to the RapidIO fabric transactions comprising a given transaction request flow in the same order that the transactions were received from the transport layer of the processing element.			

5.11A2	The physical layer of an end point processing element port shall ensure that a higher priority request transaction that it receives from the transport layer of the processing element before a lower priority request transaction with the same sourceID and the same destinationID is forwarded to the fabric before the lower priority transaction.			
5.11A3	The physical layer of an end point processing element port shall deliver transactions to the transport layer of the processing element in the same order that the packetized transactions were received by the port.			
5.11B	The device is compliant to packet delivery ordering rules			
5.11B1	A packet initiated by a processing element shall not be considered committed to the RapidIO fabric and does not participate in the packet delivery ordering rules until the packet has been accepted by the device at the other end of the link.			
5.11B2	A switch shall not alter the priority, critical request flow or VC of a packet.		Only applicable to switches	Inter-Op
5.11B3	Packet forwarding decisions made by a switch processing element shall provide a consistent output port selection which is based solely on the value of the destinationID field carried in the packet.		Only applicable to switches	Inter-Op
5.11B4	A switch processing element shall not change the order of packets comprising a transaction request flow (packets with the same sourceID, the same destinationID, the same priority, same critical request flow, same VC bit, and ftype != 8) as the packets pass through the switch.		Only applicable to switches	Inter-Op
5.11B5	A switch processing element shall not allow lower priority non-maintenance packets (ftype != 8) to pass higher priority non-maintenance packets with the same sourceID and destinationID as the packets pass through the switch.		Only applicable to switches	Inter-Op
5.11B6	A switch processing element shall not allow a priority N maintenance packet (ftype = 8) to pass another maintenance packet of priority N or greater that takes the same path through the switch (same switch input port and same switch output port).		Only applicable to switches	Inter-Op
5.11C	The device is compliant to rules for scheduling among VCs.		Only applicable if a device supports multiple VCs	Inter-Op
5.11.C1	Each VC is configured to have guaranteed bandwidth.		Only applicable if a device supports multiple VCs	Inter-Op
5.11.C1A	If the total guaranteed bandwidth for all the supported VCs is less than or equal to 100%, demand for more than its guaranteed bandwidth shall not cause any other VCs to receive less than their guaranteed bandwidth.		Only applicable if a device supports multiple VCs	Inter-Op
5.11.C2	If VC0 participates in the bandwidth reservation process, then all VCs will receive their expected minimum bandwidth.		Only applicable if a device supports multiple VCs and VC0 participates in the bandwidth reservation process.	Inter-Op

5.11.C3	If VC0 is treated with strict priority, getting whatever bandwidth is required when it has traffic to transport, the remaining VCs will divide up whatever portion of bandwidth remains.		Only applicable if a device supports multiple VCs and VC0 is treated as strict priority	Inter-Op
5.11.C4	Chapter 6, "LP-Serial Registers" defines a standard control register should the implementer decide to make the VC0 bandwidth allocation scheme a programmable feature.		Only applicable if a device supports multiple VCs and VC0 bandwidth control is programmable	Inter-Op
5.12	Deadlock Avoidance	Part 6, Section 5.12		
5.12A	Request transactions requiring responses shall only use VC0.			Inter-op
5.12B	The response packet shall only use VC0.			Inter-op
5.12C	The following requirements apply to prioritized traffic within VC0.			
5.12C1	Switch processing elements are not required, with the sole exception of ftype 8 maintenance transactions, to discern between packet types, their functions or their interdependencies.			
5.12C2	A response packet (a packet carrying a response transaction) is always assigned an initial priority, one priority level greater than the priority of the associated request packet. This requirement is specified in Table 5-2 and Table 5-3.			Inter-op
5.12C3	The end point processing element that is the source of the response packet may additionally raise the priority of the response packet to a priority higher than the minimum required by Table 5-2 and Table 5-3 if necessary for the packet to be accepted by the connected device.			
5.12C4	An end point processing element may promote a response packet only to the degree necessary for the packet to be accepted by the connected device.			
5.12D	A RapidIO fabric shall be dependency cycle free for all operations that do not require a response.			
5.12E	A packet carrying a request transaction that requires a response shall not be issued at the highest priority.			Inter-op
5.12F	A packet carrying a response shall have a priority at least one priority level higher than the priority of the associated request.			
5.12G	A switch processing element port shall accept an error-free packet of priority N if there is no packet of priority greater than or equal to N that was previously received by the port and is still waiting in the switch to be forwarded.			
5.12G1	A switch processing element port must have at least as many maximum length packet input buffers as there are priority levels.			Inter-op
5.12G2	A minimum of one maximum length packet input buffer must be reserved for each priority level. A input buffer reserved for priority N might be restricted to only priority N packets or might be allowed to hold packets of priority greater than or equal to N, either approach complies with the rule.			

5.12H	A switch processing element port that transmits a priority N packet that is forced to retry by the connected device shall select a packet of priority greater than N, if one is available, for transmission.			
5.12I	An end point processing element port shall accept an error-free packet of priority N if the port has enough space for the packet in the input buffer space of the port allocated for packets of priority N.			
5.12J	Lack of input buffer space is the only reason an end point may retry a packet.			Inter-op
5.12K	The decision of an end point processing element to accept or retry an error-free packet of priority N shall not depend on the ability of the end point to issue request packets of priority less than or equal to N from any of its ports.			
5.12K1	A port may not fill all of its buffers that can be used to hold packets awaiting transmission with packets carrying request transactions.			
5.12K2	A port must have a way of preventing output blockage at priority less than or equal to N, due to congestion in the connected device, from resulting in a lack of input buffer space for inbound packets of priority greater than or equal to N.			
5.13	Error Detection and Recovery	Part 6, Sec. 5.13		
5.13A	The CRC carried in a maintenance packet must be regenerated at each switch as the hop count changes.		Switches only	
5.13.1	Lost Packet Detection	Part 6, Sec. 5.13.1		Inter-op
5.13.1A	The RapidIO specifications require timeout counters for the physical layer, the port link timeout counters. The physical layer timeout occurs between the transmission of a packet and the receipt of an acknowledgment control symbol.			
5.13.1B	The RapidIO specifications require timeout counters for the logical layer, the port response timeout counters. The logical layer timeout occurs between the issuance of a request packet that requires a response packet and the receipt of that response packet. This timeout is counted from the time that the logical layer issues the packet to the physical layer to the time that the associated response packet is delivered from the physical layer to the logical layer.			
5.13.1C	Certain GSM operations may require two response transactions, and both must be received for the operation to be considered complete. In the case of a device implementation with multiple links, one response packet may be returned on the same link where the operation was initiated and the other response packet may be returned on a different link. If this behavior is supported by the issuing processing element, the port response timeout implementation must look for both responses, regardless on which links they are returned.		Globally shared memory support only	
5.13.2	Link Behavior Under Error	Part 6, Sec. 5.13.2		

5.13.2A	The packet transmission protocol requires that each RT packet transmitted by a port be acknowledged by the receiving port and that a port retain a copy of each RT packet that it transmits until the port receives a packet-accepted control symbol acknowledgment for the packet or the sending port determines that the packet has encountered an unrecoverable error.			
5.13.2B	If the receiving port detects a transmission error in a packet, the port sends a packet-not-accepted control symbol acknowledgment back to the sender indicating that the packet was corrupted as received.			
5.13.2C	After a link-request/input-status and link-response control symbol exchange, the sender begins retransmission with the next packet according to the priority/bandwidth scheduling rules.			
5.13.2D	The RT VCs retransmit all packets that were unacknowledged at the time of the error. CT VCs continue with the next untransmitted packet.		Multiple VCs only	
5.13.2E	All RT packets corrupted in transmission are retransmitted.			
5.13.2.1	Recoverable Errors	Part 6, Sec. 5.13.2.1		
5.13.2.1A	The following five basic types of errors are detected by a LP-Serial port:			
5.13.2.1A1	An Idle sequence error			
5.13.2.1A2	A control symbol error			
5.13.2.1A3	A packet error			
5.13.2.1A4	A column padding error			
5.13.2.1A5	A timeout waiting for an acknowledgement or link-response control symbol			
5.13.2.2	Idle Sequence Errors	Part 6, Sec. 5.13.2.2		
5.13.2.2.1	IDLE1 Sequence Errors	Part 6, Sec. 5.13.2.2.1		
5.13.2.2.1A	If an input port detects an invalid character or any valid character other then A, K, or R in an IDLE1 sequence and the port is not in the Input Error-stopped state, the port shall immediately enter the Input Error-stopped state and follow the Input Error-stopped recovery process specified in Section 5.13.2.6, "Input Error-Stopped Recovery Process".			
5.13.2.2.2	IDLE2 Sequence Errors	Part 6, Sec. 5.13.2.2.2		
5.13.2.2.2A	If an input port detects any of the following errors in an IDLE2 sequence and the port is not in the Input Error-stopped state, the port shall immediately enter the Input Error-stopped state and follow the Input Error-stopped recovery process specified in Section 5.13.2.6, "Input Error-Stopped Recovery Process".			
5.13.2.2.2A1	An invalid character or any special character other than A, K, M or R			
5.13.2.2.2A2	After lane alignment is achieved, a column that contains an A, but is not all As.			
5.13.2.2.2A3	After lane alignment is achieved, a column that contains a K, but is not all Ks.			

5.13.2.2.2A4	After lane alignment is achieved, a column that contains a M, but is not all Ms.		IDLE2 support only	
5.13.2.2.2A5	After lane alignment is achieved, a column that contains a R, but is not all Rs.			
5.13.2.2.2A6	After lane alignment is achieved, a column that contains a data character, but is not all data characters.			
5.13.2.3	Control Symbol Errors	Part 6, Sec. 5.13.2.3		
5.13.2.3.1	Link Protocol Violations	Part 6, Sec. 5.13.2.3.1		
5.13.2.3.1A	The reception of a control symbol with no detected corruption that violates the link protocol shall cause the receiving port to immediately enter the appropriate Error-stopped state.			
5.13.2.3.1B	Stype1 control symbol protocol errors shall cause the receiving port to immediately enter the Input Error-stopped state if not already in the Input Error-stopped state and follow the Input Error-stopped recovery process specified in Section 5.13.2.6, "Input Error-Stopped Recovery Process".			
5.13.2.3.1C	Stype0 control symbol protocol errors shall cause the receiving port to immediately enter the Output Error-stopped state if not already in the Output Error-stopped state and follow the Output Error-stopped recovery process specified in Section 5.13.2.7, "Output Error-Stopped Recovery Process".			
5.13.2.3.1D	If both stype0 and stype1 control symbols contain protocol errors, then the receiving port shall enter both Error-stopped states and follow both error recovery processes.			
5.13.2.3.1E	Link protocol violations include the following:			
5.13.2.3.1E1	Unexpected packet-accepted, packet-retry, or packet-not-accepted control symbol.			
5.13.2.3.1E2	Packet acknowledgment control symbol with an unexpected packet_ackID value.			
5.13.2.3.1E3	Link timeout while waiting for an acknowledgment or link-response control symbol.			
5.13.2.3.1E4	Receipt of a packet-retry symbol when operating in multi-VC mode.			
5.13.2.3.1F	The following does not constitute a protocol violation: • Receipt of a VC_status symbol when operating in single VC mode. Unexpected VC_status symbols are discarded.			
5.13.2.3.2	Corrupted Control Symbols	Part 6, Sec. 5.13.2.3.2		
5.13.2.3.2A	The reception of a control symbol with detected corruption shall cause the receiving port to immediately enter the Input Error-stopped state if not already in the Input Error-stopped state and follow the Input Error-stopped recovery process specified in Section 5.13.2.6, "Input Error-Stopped Recovery Process".			
5.13.2.3.2B	Input ports detect the following types of control symbol corruption.			
5.13.2.3.2B1	A control symbol containing invalid characters or valid but non-data characters.			

5.13.2.3.2B2	A control symbol with an incorrect CRC value.			
5.13.2.3.2B3	A control symbol whose start delimiter (SC or PD) occurs in a lane whose lane_number mod4 != 0.			
5.13.2.3.2B4	A long control symbol that does not have a end delimiter in the seventh character position after its start delimiter and with the same value as the start delimiter.			
5.13.2.4	Packet Errors	Part 6, Sec. 5.13.2.4		Inter-op
5.13.2.4A	Each packet received by a port shall be checked for the following types of errors:			
5.13.2.4A1	Packet with an unexpected ackID value.			
5.13.2.4A2	Packet with an incorrect CRC value.			
5.13.2.4A3	Packet containing invalid characters or valid non-data characters.			
5.13.2.4A4	Packet that exceeds the maximum packet size (276 bytes).			
5.13.2.4B	With one exception, the reception of a packet with any of the above errors shall cause the receiving port to immediately enter the Input Error-stopped state if not already in the Input Error-stopped state and follow the Input Error-stopped recovery process specified in Section 5.13.2.6, "Input Error-Stopped Recovery Process".			
5.13.2.4c	The exception occurs when the link to which the port is connected is operating with the IDLE2 idle sequence, the packet in which one or more errors were detected was canceled by a link-request control symbol, and the only errors detected in the packet were the presence of one or more M special characters and may cause excessive packet length. In this case, the errors detected in the packet shall be ignored and the packet handled as a canceled packet as specified in Section 5.10, "Canceling Packets".			
5.13.2.5	Link Timeout	Part 6, Sec. 5.13.2.5		Inter-op
5.13.2.5A	A link timeout while waiting for an acknowledgment or link-response control symbol is handled as a link protocol violation as described in Section 5.13.2.3.1, "Link Protocol Violations".			
5.13.2.6	Input Error-Stopped Recovery Process	Part 6, Sec. 5.13.2.6		Inter-op
5.13.2.6A	When the input side of a port detects a transmission error, it immediately enters the Input Error-stopped state.			
5.13.2.6B	To recover from this state, the input side of the port takes the following actions.			
5.13.2.6B1	Record the condition(s) that caused the port to enter the Input Error-stopped state.			
5.13.2.6B2	If an error(s) was detected in a control symbol or packet, ignore and discard the corrupted control symbol or packet.			
5.13.2.6B3	Cause the output side of the port to issue a packet-not-accepted control symbol.			

5.13.2.6B4	Subsequent to the event that caused the port to enter the Input Error-stopped state and prior to the reception of a link-request/input-status control symbol, discard without acknowledgement or error report all packets that are received for VCs operating in RT mode.			
5.13.2.6B5	Subsequent to the event that caused the port to enter the Input Error-stopped state and prior to the reception of a link-request/input-status control symbol, accept without acknowledgement (accept silently) all error free packets that are received for VCs operating in CT mode for which the VC specified in the packet has buffer space available.		Multiple VCs with CT support only	
5.13.2.6B6	Subsequent to the event that caused the port to enter the Input Error-stopped state and prior to the reception of a link-request/input-status control symbol, discard without acknowledgement all packets that are received for VCs operating in CT mode which are not error free or for which the VC specified in the packet does not have buffer space available.			
5.13.2.6B7	When a link-request/input-status control symbol is received from the connected port, cause the output side of the port to transmit a link-response control symbol and if the transmitter-controlled flow control is in use on the link, to also transmit a VC_Status control symbol for each of VC1-8 that is active. The transmission of a VC_Status control symbol for each of VC1-8 that is active is optional if receiver-controlled flow control in use on the link.			
5.13.2.6B8	The input side of the port should also cause the output side of the port to transmit a status control symbol (for VC0).			
5.13.2.6B9	The input side of the port then exits the Input Error-stopped state and resumes normal packet reception.			
5.13.2.6C	The transmission of the link-response, status and VC-status control symbols is subject to the following requirements.			
5.13.2.6C1	The link-response control symbol shall be transmitted either before any of the status and VC-status control symbols are transmitted or after all of the status and VC-status control symbols are transmitted.			
5.13.2.6C2	The status and VC-status control symbols that are transmitted shall be transmitted in the following order. If a status control symbol is transmitted it shall be transmitted first before any of the VC-status control symbols. Any VC-status control symbols that are transmitted shall be transmitted after the status control symbol and in order of increasing VCID.			

5.13.2.6C3	The link-response control symbol shall not be transmitted until the input side of the port is ready to resume packet reception and either the buffer consumption of all packets received by the port before the link-request/input-status control symbol has been determined or the port is able to maintain the distinction after packet reception resumes between packets received before the reception of the link-request/input-status control symbol and packets received after the reception of the link-request/input-status control symbol.			
5.13.2.6C4	The status or VC-status control symbol for a VC operating in RT mode shall indicate the number of receive buffers available for that VC inclusive of the buffer consumption of all packets received and accepted by the port for that VC before the event that caused the port entered the Input Error-stopped state.			
5.13.2.6C5	The VC-status control symbol for a VC operating in CT mode shall indicate the number of receive buffers available for that VC inclusive of the buffer consumption of all packets received and accepted by the port for that VC before the link-request/input-status control symbol was received.		Multiple VCs with CT support only	
5.13.2.6C6	The status and VC-status control symbols shall be transmitted before any packet acknowledgment control symbols are transmitted for packets received after the link-request/input-status control symbol was received.			
5.13.2.7	Output Error-Stopped Recovery Process	Part 6, Sec. 5.13.2.7		Inter-op
5.13.2.7A	To recover from the Output Error-stopped state, the output side of a port takes the following actions.			
5.13.2.7A1	Immediately stops transmitting new packets.			
5.13.2.7A2	Resets the link packet acknowledgment timers for all transmitted but unacknowledged packets.			
5.13.2.7A3	Transmits an input-status link-request/input-status (restart-from-error) control symbol.			
5.13.2.7A4	When the link-response is received, VCs operating in RT mode back up to the first unaccepted packet in each VC.			
5.13.2.7A5	VCs operating in CT mode silently assume the unacknowledged packets were accepted and adjust their state accordingly.		Multiple VCs with CT support only	
5.13.2.7A6	The port exits the Output Error-stopped state and resumes transmission with the next RT or CT packet according to the bandwidth allocation algorithm using the ackID value contained in the link-response control symbol.			

Item #	Compliance Item	Specification Reference	Optional	Interop Item
6	LP-Serial Registers	Part 6, Chap. 6		
6.2	Register Map	Part 6, Sec. 6.2		
6.2A	LP-Serial registers utilize the Extended Features blocks and can be accessed using RapidIO Part 1: Input/Output Logical Specification maintenance operations.			
6.2B	Read and write accesses to reserved register offsets shall terminate normally and not cause an error condition in the target device.			
6.2C	LP-Serial registers shall be mapped according to Table 6-1.			
6.2D	All register maps described in this session can be extended or shortened if more or less port definitions are required for a device.			
6.2E	If less than 16 ports are defined, remaining register map offset can be used for another Extended Features block.			
6.2F	Port Maintenance Block Header CSR is read-only.			
6.2G	Port Maintenance Block Header CSR has the proper Extended Features block ID in the EF_ID field.			
6.3	Reserved Register and Bit Behavior	Part 6, Sec. 6.3		
6.3A	Accesses to reserved registers and register bits shall follow Table 6-2 , this applies to all registers specified in Chapter 6.			
6.4	Capability Registers (CARs)	Part 6, Sec. 6.4		
6.4A	Every processing element shall contain a set of registers that allows an external processing element to determine its capabilities using the I/O logical maintenance read operation.			
6.4B	All registers are 32 bits wide and are organized and accessed in 32-bit (4 byte) quantities, although some processing elements may optionally allow larger accesses.			
6.4C	CARs are read-only.			
6.4D	CARs are big-endian with bit 0 the most significant bit.			
6.4.1	Processing Element Features CAR	Part 6, Sec. 6.4.1		
6.4.1A	Bit Settings of Processing Element Features CAR shall be mapped as in table 6-3.			
6.4.1B	The multiport bit shall be implemented by devices that support the LP-Serial IDLE2 sequence, but is optional for devices that do not support the LP-Serial IDLE2 sequence. If this bit is not implemented it is Reserved.		Required if idle2 is supported	
6.4.1C	If the multiport bit is implemented, the Switch Port Information CAR at Configuration Space Offset 0x14 shall be implemented regardless of the state of bit 3 of the Processing Element Features CAR.		Required if idle2 is supported	
6.4.1D	The CRF bit should be set to 1 if PE supports CRF, otherwise set it to 0.			
6.5	LP-Serial Extended Feature Blocks	Part 6, Sec. 6.5		
6.5A	All registers in the LP-Serial Extended Feature Blocks are 32 bits in length and aligned to 32 bit boundaries.			

6.5B	All registers in the 1x/4x/LP-Serial Physical Layer Specification Extended Features data structure can be read and/or written, as per constraints specified in Part 6, Chapter 6. The Extended Features Space is located at bytes offsets 0x0100 through 0xFFFC of the device configuration space.	Part 6, Sec. 6.2		
6.5.1	Generic End Point Devices	Part 6, Sec. 6.5.1	Required for generic end point device	
6.5.1A	All register maps described in this session can be extended or shortened if more or less port definitions are required for a device.			
6.5.1B	This Extended Features register block is assigned Extended Features block ID=0x0001.			
6.5.1C	Registers of the RapidIO LP-Serial Extended Features Block for generic end point devices shall be mapped according to Table 6-4.			
6.5.1D	If less than 16 ports are defined, remaining register map offset can be used for another Extended Features block.			
6.5.1E	The Block Offset is the offset relative to the 16-bit Extended Features Pointer (EF_PTR) that points to the beginning of the block. The address of a byte in the block is calculated by adding the block byte offset to EF_PTR that points to the beginning of the block.			
6.5.2	Generic End Point Devices, software assisted error recovery option	Part 6, Sec. 6.5.2	Required for end point device with software assisted error recovery option	
6.5.2A	This Extended Features register block is assigned Extended Features block ID=0x0002.			
6.5.2B	Registers generic RapidIO LP-Serial end point devices with software assisted error recovery shall be mapped according to Table 6-5.			
6.5.2C	All register maps described in this session can be extended or shortened if more or less port definitions are required for a device.			
6.5.2D	If less than 16 ports are defined, remaining register map offset can be used for another Extended Features block.			
6.5.3	Generic End Point Free Devices	Part 6, Sec. 6.5.3	Required for generic end point free device	
6.5.3A	This Extended Features register block uses extended features block ID=0x0003.			
6.5.3B	Registers of generic RapidIO LP-Serial end point-free devices shall be mapped according to Table 6-6.			
6.5.3C	The Block Offset is the offset relative to the 16-bit Extended Features Pointer (EF_PTR) that points to the beginning of the block. The address of a byte in the block is calculated by adding the block byte offset to EF_PTR that points to the beginning of the block.			
6.5.3D	All register maps described in this session can be extended or shortened if more or less port definitions are required for a device.			
6.5.3E	If less than 16 ports are defined, remaining register map offset can be used for another Extended Features block.			

6.5.4	Generic End Point Free Devices, software assisted error recovery option	Part 6, Sec. 6.5.4	Required for end point free device with software assisted error recovery option	
6.5.4A	This Extended Features register block is assigned Extended Features block ID=0x0009.			
6.5.4B	Registers of generic RapidIO LP-Serial end point-free devices with software assisted error recovery shall be mapped according to Table 6-7.			
6.5.4C	The Block Offset is the offset relative to the 16-bit Extended Features Pointer (EF_PTR) that points to the beginning of the block. The address of a byte in the block is calculated by adding the block byte offset to EF_PTR that points to the beginning of the block.			
6.5.4D	All register maps described in this session can be extended or shortened if more or less port definitions are required for a device.			
6.5.4E	If less than 16 ports are defined, remaining register map offset can be used for another Extended Features block.			
6.6	LP-Serial Command and Status Registers (CSRs)	Part 6, Sec. 6.6		
6.6A	All Command and Status registers are 32 bits in length and are aligned to 32 bit boundaries. All CSRs are accessed as 4 byte entities.			
6.6B	CSRs are big endian with bit 0 the most significant bit.			
6.6.1	LP-Serial Register Block Header	Part 6, Sec. 6.6.1		
6.6.1A	Block offset for LP-Serial Register Block Header is 0x0.			
6.6.1B	The LP-Serial register block header register contains the EF_PTR to the next extended features block and the EF_ID that identifies LP-Serial Extended Feature Block for which this is the register block header.			
6.6.1C	Bit Settings of LP-Serial Register Block Header shall be implemented according to Part 6, Table 6-8.			
6.6.2	Port Link Timeout Control CSR	Part 6, Sec. 6.6.2		
6.6.2A	Block offset for Port Link Timeout Control CSR is 0x20.			
6.6.2B	The port link timeout control register contains the timeout timer value for all ports on a device.			
6.6.2C	The port link timeout is for link events such as sending a packet to receiving the corresponding acknowledge and sending a link-request to receiving the corresponding link-response.			
6.6.2D	The port link timeout control register reset value is the maximum timeout interval (all 1s), and represents between 3 and 6 seconds.			
6.6.2E	Bit Settings of Port Link Timeout Control CSR shall be implemented according to Part 6, Table 6-9.			
6.6.3	Port Response Timeout Control CSR	Part 6, Sec. 6.6.3	Required for end point device	
6.6.3A	Block offset for Port Response Timeout Control CSR is 0x24.			
6.6.3B	The port response timeout control register contains the timeout timer count for all ports on a device.			

6.6.3C	Port Response Timeout is for sending a request packet to receiving the corresponding response packet.			
6.6.3D	The port response timeout control register reset value is the maximum timeout interval (all 1s), and represents between 3 and 6 seconds.			
6.6.3E	Bit Settings of Port Response Timeout Control CSR shall be implemented according to Part 6, Table 6-10.			
6.6.4	Port General Control CSR	Part 6, Sec. 6.6.4		
6.6.4A	Block offset for Port General Control CSR is 0x3C.			
6.6.4B	The bits accessible through the Port General Control CSR are bits that apply to all ports in a device.			
6.6.4C	Bit settings of Port General Control CSR for generic end point device shall be implemneted according to Part 6, Table 6-11.			
6.6.4D	Bit settings for General Port Control CSR For generic end point free device shall be implemented according to Part 6, Table 6-12.			
6.6.4E	Discovered field shall set to 1 as soon as a maintenance write request is received to this register bit and Discovered bit shall be 0 after reset for this device.	sRIO1.3 checklist		
6.6.5	Port n Link Maintenance Request CSRs	Part 6, Sec. 6.6.5	Required if software assisted error recovery is supported	
6.6.5A	A write to one of these registers generates a link-request control symbol on the corresponding RapidIO port interface.			
6.6.5B	Bit settings of Port n Link Maintenance Request CSRs shall follow Part 6, Table 6-13.			
6.6.5C	The reset value of the command field is 3'b000.			
6.6.5D	If the field is read, it returns the last written value.			
6.6.5E	The reset value of the Port n Link Maintenance Request CSR is 0.			
6.6.6	Port n Link Maintenance Response CSRs.	Part 6, Sec. 6.6.6	Required if software assisted error recovery is supported	
6.6.6A	A read to Port n Link Maintenance Response CSRs returns the status received in a link-response control symbol.			
6.6.6B	Port n Link Maintenance Response CSR is read-only.			
6.6.6C	Bit settings of Port n Link Maintenance Response CSRs shall follow Part 6, Table 6-14.			
6.6.6D	Response_valid field indicates that the link-response has been received and the status fields are valid. If the link-request does not cause a link-response, this bit indicates that the link-request has been transmitted.			
6.6.6E	Response_valid field clears on read.			
6.6.6F	The reset value of the Port n Link Maintenance Response CSR register is 0.			
6.6.7	Port n Local ackID CSRs	Part 6, Sec. 6.6.7	Required if software assisted error recovery is supported	
6.6.7A	Block offset for Port n Local ackID CSRs starts from 0x48, add 0x20 for each additional port.			

6.6.7B	A read to Port n Local ackID CSRs returns the local ackID status for both the output and input sides of the ports.			
6.6.7C	Bit settings for Port n Local ackID Status CSRs shall follow Part 6, Table 6-15.			
6.6.7D	Writing 0b1 toClr_outstanding_ackIDs bit causes all outstanding unacknowledged packets to be discarded.			
6.6.7E	The Clr_outstanding_ackIDs bit is always logic 0 when read.			
6.6.7F	Outstanding_ackID field indicates the ackID value expected in the next received acknowledge control symbol.			
6.6.7G	Outbound_ackID field indicates output port next transmitted ackID value. Software writing this field can force retransmission of outstanding unacknowledged packets in order to manually implement error recovery.			
6.6.7H	Port n Local ackID CSRs reset value is 0x0			
6.6.8	Port n Error and Status CSRs	Part 6, Sec. 6.6.8		
6.6.8A	Block offset of Port n Error and Status CSRs starts from 0x58, add 0x20 for each additional port.			
6.6.8B	Bit settings of Port n Error and Status CSRs shall follow Part 6, Table 6-16.			
6.6.8C	"Idle Sequence 2 Enable" bit should reset to "1" if "idle Sequence2 Support" bit is set, otherwise reset to "0".			
6.6.8D	The port shall not allow the "Idle Sequence 2 Enable" bit to be set unless idle sequence 2 is supported and shall not allow this bit to be cleared if only idle sequence 2 is supported.			
6.6.8E	All fields defined in this register are read only except the "Idle Sequence 2 Enable" bit which is R/W.			
6.6.8F	"Output Retry-encountered" bit is set when "output retry-stopped" bit is set. Once set, remains set until written with a logic 1 to clear.			
6.6.8G	"Output Retried" bit is set when "output retry-stopped" bit is set and is cleared when a packet-accepted or a packet-not-accepted control symbol is received.			
6.6.8H	"Output Error-encountered" bit is set when "Output Error-stopped" bit is set. Once set, remains set until written with a logic 1 to clear.			
6.6.8I	"Input Error-encountered" bit is set when "Input Error-stopped" bit is set. Once set, remains set until written with a logic 1 to clear.			
6.6.8J	"Port-write Pending" bit is only valid if the device is capable of issuing a maintenance port-write transaction. Once set remains set until written with a logic 1 to clear.			
6.6.8K	"Port Error" bit indicates Input or output port has encountered an error from which hardware was unable to recover. Once set, remains set until written with a logic 1 to clear.			
6.6.9	Port n Control CSRs	Part 6, Sec. 6.6.9		
6.6.9A	Block offset of Port n Control CSRs starts from 0x5C, add 0x20 for each additional port.			
6.6.9B	Bit settings of Port n Control CSRs shall follow Part 6, Table 6-17.			

6.6.9C	"Port Width Support" field is read-only field. It indicates port width modes supported by the port.			
6.6.9D	"Initialized Port Width" indicates width of the ports after initialized. It is read-only.			
6.6.9E	The port shall not allow the enabling of a width mode that is not supported by the port.			
6.6.9F	A change in the value of the "Port Width Override" and "Extended port width override" field shall cause the port to re-initialize using the new field value.			
6.6.9G	If output port enable bit is not set, port is stopped and not enabled to issue any packets except to route or respond to I/O logical MAINTENANCE packets.Control symbols are not affected and are sent normally.			
6.6.9H	If input port enable bit is not set, port is stopped and only enabled to route or respond I/O logical MAINTENANCE packets. Control symbols are not affected and are sent normally.			
6.6.9I	If input port enable bit is not set, the port generates packet-not-accepted control symbols to all received non-maintenance packets.			
6.6.9J	An enumeration boundary aware system enumeration algorithm shall honor the "Enumeration Boundary" flag. The algorithm, on either the ingress or the egress port, shall not enumerate past a port with this bit set.			
6.6.9K	"Extended Port Width Override" is used and defined in conjunction with the bits in the Port Width Override field.			
6.6.9L	"Extended Port Width Support" field indicates additional port width modes supported by the port (read-only).			
6.6.9M	The Enumeration Boundary reset value is implementation dependent; provision shall be made to allow the reset value to be configurable if this feature is supported.			
6.6.10	Port n Control 2 CSRs	Part 6, Sec. 6.6.10		
6.6.10A	Block offset of Port n Control CSRs starts from 0x54, add 0x20 for each additional port.			
6.6.10B	Bit settings of Port n Control 2 CSRs shall follow Part 6, Table 6-18.			
6.6.10C	The port shall not allow "Baudrate Discovery Enable" bit to be set unless it supports baudrate discovery.			
6.6.10D	The port shall not allow the "BAUD Enable" bits to be set unless it supports the specified Baud.			
6.6.10E	Enable Inactive Lanes bit		Optional	
6.6.10E1	The implementation of this bit is optional. When implemented, this bit allows software to force the lanes of the port that are not currently being used to carry traffic, the “inactive lanes”, to be enabled for testing while the “active lanes” continue to carry traffic. If this bit is not implemented it is reserved.			
6.6.10E2	When a 1x/Nx or 1x/Mx/Nx port is operating in 1x mode where 1<M<N and N = 4, 8 or 16, lanes 0 and 2 are the active lanes and lane 1 and lanes 3 through N-1 are the inactive lanes.			

6.6.10E3	When a 1x/Mx/Nx port is operating in Mx mode where 1<M<N and N = 4, 8 or 16, lanes 0 through M-1 are the active lanes and lanes M through N-1 are the inactive lanes.			
6.6.10E4	Use of the test mode enabled by the implementation of the "Enable Inactive Lanes" bit to monitor the behavior of the inactive lanes requires that this bit must be set in both ports and that all link width modes wider than the desired Mx mode must be disabled in the Port n Control CSR of both ports.		Optional	
6.6.10E5	If implemented, this bit shall not be asserted when the port is connected to a link that includes retimers as defined in Section 4.11.1, "Retimers".		Optional	
6.6.10E6	The port's drivers for the inactive lanes are output enabled if and only if the port's Initialization state machine is not in the SILENT or SEEK state.		Optional	
6.6.10E7	A continuous IDLE sequence of the same type as is in use on the active lanes shall be transmitted on the inactive lanes when their transmitters are output enabled.		Optional	
6.6.10E8	The IDLE sequences transmitted on the inactive lanes shall comply with all rules for that type of IDLE sequence including alignment across the inactive lanes, but they are not required to use the same bit sequences or be aligned in any way relative to the IDLE sequences transmitted on the active lanes.		Optional	
6.6.10E9	If IDLE2 is being used on the active lanes of the port, the inactive lanes of the port shall report their lane number and port width in the CS Field Marker and handle commands carried in the CS Field as if they were active lanes.		Optional	
6.6.10F	When the "Data scrambling disable" bit is set, the transmit scrambler and receive descrambler are disabled for control symbol and packet data characters, but the transmit scrambler remains enabled for the generation of pseudo-random data characters for the IDLE2 random data field.		Optional	
6.6.10G	The port shall not let the "Remote Transmit Emphasis Control Enable" bit be set unless remote transmit emphasis control is supported and the link to which the port is connect is using idle sequence 2 (IDLE2).			
6.6.10H	The "1.25G/2.5G/3.125G/5.0G/6.25 G Baud Eanble" bits shall reset to "1" if its corresponding "Baud Support" bits are set, otherwise reset to 0.			
6.7	LP-Serial Lane Extended Features Block	Part 6, Sec. 6.7		
6.7A	All registers in this block are 32 bits in length, aligned to a 32-bit (4-byte) boundary and accessed as 4 byte entities, although some processing elements may optionally allow larger accesses.			
6.7B	This Extended Features register block is assigned Extended Features block ID=0x000D.			
6.7.1	Register Map	Part 6, Sec. 6.7.1		
6.7.1A	Registers of the RapidIO LP-Serial Lane Extended Features Block shall be mapped according to Part 6, Table 6-19.			

6.7.1B	The Block Offset is the offset relative to the 16-bit Extended Features Pointer (EF_PTR) that points to the beginning of the block. The address of a byte in the block is calculated by adding the block byte offset to EF_PTR that points to the beginning of the block.			
6.7.1C	All register maps described in this session can be extended or shortened if more or less port definitions are required for a device.			
6.7.1D	If less than 16 ports are defined, remaining register map offset can be used for another Extended Features block.			
6.7.2	LP-Serial Lane Command and Status Registers (CSRs)	Part 6, Sec. 6.7.2		
6.7.2.1	LP-Serial Register Block Header	Part 6, Sec. 6.7.2.1		
6.7.2.1A	The LP-Serial register block header register contains the EF_PTR to the next extended features block and the EF_ID that identifies LP-Serial Lane Extended Feature Block for which this is the register block header.			
6.7.2.1B	Bit settings for LP-Serial Register Block Header shall follow Part 6, Table 6-20.			
6.7.2.2	Lane n Status 0 CSRs	Part 6, Sec. 6.7.2.2		
6.7.2.2A	Block offset of Lane n Status 0 CSRs starts from 0x10, add 0x20 for each lane increment.			
6.7.2.2B	Bit settings of Lane n Status 0 CSRs shall follow Part 6, Table 6-21.			
6.7.2.2C	Unless otherwise specified, all bits in this register are read-only (RO).			
6.7.2.2D	If the lane supports the IDLE2 sequence, the value of the "Receiver trained" bit shall be the same as the value in the “Receiver trained” bit in the CS Field transmitted by the lane.			
6.7.2.2E	The "8B/10B decoding errors" field indicates the number of 8B/10B decoding errors that have been detected for this lane since this register was last read. The field is reset to 0x0 when the register is read.			
6.7.2.2F	The "Lane_sync state change" bit indicates whether the lane_sync signal for this lane has changed state since the bit was last read. This bit is reset to 0b0 when the register is read.			
6.7.2.2G	The "Rcvr_trained state change" indicates whether the rcvr_trained signal for this lane has changed state since the bit was last read. This bit is reset to 0b0 when the register is read.			
6.7.2.3	Lane n Status 1 CSRs	Part 6, Sec. 6.7.2.3	Required if IDLE2 supported	
6.7.2.3A	Block offset of Lane n Status 1 CSRs starts from 0x14, add 0x20 for each lane increment.			
6.7.2.3B	This register shall be implemented if and only if the lane supports the IDLE2 sequence.			
6.7.2.3C	Only information from error free CS markers and CS fields shall be reported in this register.			
6.7.2.3D	Unless otherwise specified, all bits in this register are read-only (RO).			
6.7.2.3E	Bit settings of Lane n Status 1 CSRs shall follow Part 6, Table 6-22.			

6.7.2.3F	The"IDLE2 received" bit is R/W. This bit can be reset by writing the bit with the value 0b1. Writing the bit with the value 0b0 does not change the value of the bit.			
6.7.2.3G	When the "IDLE2 information current" bit is asserted, it indicates that the information is from the last IDLE2 CS Marker and CS Field that were received by the lane without detected errors, and that the lane’s lane_sync signal has remained asserted since the last CS Marker and CS Field were received.			
6.7.2.3H	The "Values changed" bit is reset when the register is read.			
6.7.2.4	Implementation Specific CSRs	Part 6, Sec. 6.7.2.4	Optional	
6.7.2.4A	The registers shall be implemented in increasing numerical order beginning with the Lane n Status 2 CSR.		Required if implementation specific CSRs are implemented	
6.7.2.4.1	Block offset of Lane n Status 2 CSRs starts from 0x18, add 0x20 for each lane increment.			
6.7.2.4.2	Block offset of Lane n Status 3 CSRs starts from 0x1C, add 0x20 for each lane increment.			
6.7.2.4.3	Block offset of Lane n Status 4 CSRs starts from 0x20, add 0x20 for each lane increment.			
6.7.2.4.4	Block offset of Lane n Status 5 CSRs starts from 0x24, add 0x20 for each lane increment.			
6.7.2.4.5	Block offset of Lane n Status 6 CSRs starts from 0x28, add 0x20 for each lane increment.			
6.7.2.4.6	Block offset of Lane n Status 7 CSRs starts from 0x2c, add 0x20 for each lane increment.			
6.8	Virtual Channel Extended Features	Part 6, Sec. 6.8	Required if VCs are supported	
6.8A	This Extended Features register block is assigned Extended Features block EF_ID=0x000A.			
6.8.1	Register Map	Part 6, Sec. 6.8.1		
6.8.1A	The virtual channel registers for RapidIO LP-Serial devices shall be mapped according to Part 6, Table 6-23.			
6.8.1.B	The default method is to configure VC operation when the channel is quiescent either by protocol, or by holding the master enable in the disabled state.			
6.8.2	Virtual Channel Control Block Registers	Part 6, Sec. 6.8.2	Required if VCs are supported	
6.8.2.1	VC Register Block Header	Part 6, Sec. 6.8.2.1		
6.8.2.1A	Block offset of the "VC Register Block Header" register is 0x0.			
6.8.2.1B	The LP-Serial VC register block header register contains the EF_PTR to the next extended features block and the EF_ID that identifies this as the Virtual Channel Extended Features Block.			
6.8.2.1A	Bit settings of VC Register Block Header shall follow Part 6, Table 6-24.			
6.8.2.2	Port n VC Control and Status Registers	Part 6, Sec. 6.8.2.2		
6.8.2.2A	Block Offset of the "Port n VC Control and Status Registers" starts from 0x20, add 0x20 for each port increament.			
6.8.2.2B	Bit settings of Port n VC Control and Status Registers shall follow Part 6, Table 6-25.			
6.8.2.2C	Implementers are required to support a maximum VC refreshing period of at least 1024 x 16 = 16K code groups in size.			

6.8.2.2D	The maximum possible VC refreshing period that can be supported is 1024 x 256 = 256K code groups. Writing to this field with a value greater than the maximum supported value by the port will set the field to the maximum value supported by the port.			
6.8.2.2E	CT mode must be implemented in the highest VCs first to allow this simplified programming model.			
6.8.2.2F	VCs not supporting CT operation are indicated by not allowing the "CT Mode" bits to set.			
6.8.2.2G	"VCs Enable" Bits 24-27, and any bits associated with unimplemented VCs need not be writable, but must return 0 when read.			
6.8.2.2H	Setting the "VCs Enable" field to a value larger than the number of VCs supported as indicated in "VCs Support" field. Will result in only VC0 being enabled.			
6.8.2.3	Port n VC0 BW Allocation Registers	Part 6, Sec. 6.8.2.3		
6.8.2.3A	Block Offset of the "Port n VC0 BW Allocation Registers" starts from 0x24, add 0x20 for each port increament.			
6.8.2.3B	Bit settings of Port n VC0 BW Allocation CSR shall follow Part 6, Table 6-26.			
6.8.2.3C	The "VC0 Bandwidth Reservation Capable" bit indicates whether the device supports VC0 bandwidth reservation scheduling. This bit is read only.			
6.8.2.3D	The bandwidth allocation value is left justified based on precision. Bits are ignored based on the precision value.			
6.8.2.3E	When VC0 in strict priority mode (default mode),traffic on VC0 is serviced before any of the other VCs. The remaining bandwidth is then divided according to the percentages in the bandwidth allocations.			
6.8.2.3F	Optionally, VC0 may be included in the bandwidth reservation scheduling. In this case, the priorities within VC0 are serviced when VC0 is allocated bandwidth on the link. VC0 activity cannot cause the other VCs to receive less than their allocation of bandwidth in this mode.			
6.8.2.3H	The Bandwidth Reservation Precision field is used to indicate the granularity of bandwidth scheduling for the port. The value in this register applies to the subsequent BW Allocation Registers as well.			
6.8.2.3I	The value programmed in the BW Allocation Registers is a binary fraction based on the percentage of the overall total bandwidth. 100% bandwidth is represented by a value of 1.000.			
6.8.2.3J	if the percentage results in a value smaller than the precision, a value of 0 could result in a VC getting no service. The precision value allows the bandwidth allocation algorithm to round up or down based on the dividing point, and to detect and round up a zero value to allocate at least a minimal increment of bandwidth.			
6.8.2.3K	The total of all the allocations should not exceed 100%.			
6.8.2.4	Port n VCx BW Allocation Registers	Part 6, Sec. 6.8.2.4		

6.8.2.3A	Block Offset of the "Port n VCx BW Allocation Registers" is based on VC number, it shall be mapped according to table 6-23.			
6.8.2.4A	Each Port n VCx BW Allocation CSR register supports 2 VCs. bit settings of the register shall follow Part 6, Table 6-28.			
6.8.2.4B	The bandwidth allocation value is left justified based on precision. Bits are ignored based on the precision value.			
6.8.2.4C	If the VC is not enabled, its bandwidth allocation bits are treated as reserved.			
6.8.2.4D	A value of ‘0’ for bandwidth allocation results in no service being given to that VC.			
6.8.2.4E	VCs initialize with a value of zero and remain inactive until allocated bandwidth.			

Item #	Compliance Item	Specification Reference	Optional	Interop Item
7	Signal Descriptions	Part 6, Chap. 7		
7.1	The interface is defined either as a 1x, 2x, 4x, 8x, or 16x lane, full duplex, point-to-point interface using differential signaling. A lane implementation consists of Nx4 wires with two for the egress and two for the ingress direction.	Part 6, Sec. 7.1		
7.2	See Table 7-1. LP-Serial Signal Description	Part 6, Sec. 7.2		

Item #	Compliance Item	Specification Reference	Optional	Interop Item
8	Common Electrical Specifications	Part 6, Chap. 9		
8.1	Introduction	Part 6, Sec. 8.1		
8.1A	A Level I link shall meet the Level I requirements listed in Section 8.1.		Required only for Level I links	
8.1B	A Level II link shall meet the Level II requirements listed in Section 8.1.		Required only for Level II links	
8.5.2	Links shall meet the data pattern requirements specified in Section 8.5.2.	Part 6, Sec. 8.5.2		
8.5.3	Links shall meet the signal level requirements specified in Section 8.5.3.	Part 6, Sec. 8.5.3		
8.5.4	Links shall meet the bit error ratio requirements specified in Section 8.5.4.	Part 6, Sec. 8.5.4		
8.5.5	Links shall meet the ground differences requirements specified in Section 8.5.5.	Part 6, Sec. 8.5.5		
8.5.6	Links shall meet the cross talk requirements specified in Section 8.5.6.	Part 6, Sec. 8.5.6		
8.5.7	Links shall meet the transmitter test load requirements specified in Section 8.5.7.	Part 6, Sec. 8.5.7		
8.5.8	Links shall meet the transmitter lane-to-lane skew requirements specified in Section 8.5.8.	Part 6, Sec. 8.5.8		
8.5.9	Links shall meet the receiver input lane-to-lane skew requirements specified in Section 8.5.9.	Part 6, Sec. 8.5.9		
8.5.10	Links shall meet the transmitter short circuit current requirements specified in Section 8.5.10.	Part 6, Sec. 8.5.10		
8.5.11	Links shall meet the differential resistance and return loss requirements specified in Section 8.5.11.	Part 6, Sec. 8.5.11		
8.5.12	Links shall meet the baud rate tolerance requirements specified in Section 8.5.12.	Part 6, Sec. 8.5.12		
8.5.13	Links shall meet the termination and DC blocking requirements specified in Section 8.5.13.	Part 6, Sec. 8.5.13		

Item #	Compliance Item	Specification Reference	Optional	Interop Item
9	1.25Gbaud, 2.5Gbaud, and 3.125Gbaud LP-Serial Links	Part 6, Chap. 9	Required only for Level I links	
9.1	Level I short run and long run electrical interfaces operating at 1.25Gbaud, 2.5Gbaud, and 3.125Gbaud shall meet the Level I application goals detailed in Section 9.1.	Part 6, Sec. 9.1		
9.2	The types of equalizers and, if the equalizers are adaptive, the adaptive equalizer training algorithms that may be used in Level I transmitter or receiver are subject to the restrictions detailed in Section 9.2.	Part 6, Sec. 9.2		
9.4	Level I Electrical Specification	Part 6, Sec. 9.4		
9.4.1	Level I Short Run Transmitter Characteristics	Part 6, Sec. 9.4.1		
9.4.1A	Level I short run transmitter must meet the AC timing specifications specified in Table 9-1 and detailed in Section 9.4.1.			
9.4.1B	Level I short run transmitter must meet the output jitter specifications specified in Table 9-2 and detailed in Section 9.4.1.			
9.4.2	Level I Long Run Transmitter Characteristics	Part 6, Sec. 9.4.2		
9.4.2A	Level I long run transmitter must meet the AC timing specifications specified in Table 9-5 and detailed in Section 9.4.2.			
9.4.2B	Level I long run transmitter must meet the output jitter specifications specified in Table 9-6 and detailed in Section 9.4.2.			
9.4.3	Level I Receiver Specifications	Part 6, Sec. 9.4.3		
9.4.3A	Level I receiver must meet the electrical input specifications specified in Table 9-9 and detailed in Section 9.4.3.			
9.4.3B	Level I receiver must meet the input jitter tolerance specifications specified in Table 9-10 and detailed in Section 9.4.3.			
9.5	Level I measurement and test requirements must be followed according to Section 9.5.	Part 6, Sec. 9.5		

Item #	Compliance Item	Specification Reference	Optional	Interop Item
10	5Gbaud and 6.25Gbaud LP-Serial Links	Part 6, Chap. 10	Required only for Level II links	
10.1	Level II short run, medium run, and long run electrical interfaces operating at 5Gbaud and 6.25Gbaud shall meet the Level II application goals detailed in Section 10.1.	Part 6, Sec. 9.1		
10.2	The types of equalizers and, if the equalizers are adaptive, the adaptive equalizer training algorithms that may be used in Level II 5Gbaud transmitter or receiver are subject to the restrictions detailed in Section 10.2.	Part 6, Sec. 9.2	5Gbaud only	
10.4	Level II Short Run Interface General Requirements	Part 6, Sec. 10.4	Short run only	
10.4.1	Jitter and Inter-operability Methodology	Part 6, Sec. 10.4.1		
10.4.1.1	Level II short run link must meet the defined test patterns requirements specified in Section 10.4.1.1.	Part 6, Sec. 10.4.1.1		
10.4.1.2	Level II short run link must meet the channel compliance requirements specified in Section 10.4.1.2	Part 6, Sec. 10.4.1.2		
10.4.1.3	Level II short run transmitter must meet the inter-operability requirements specified in Section 10.4.1.3.	Part 6, Sec. 10.4.1.3		
10.4.1.4	Level II short run receiver must meet the inter-operability requirements specified in Section 10.4.1.4.	Part 6, Sec. 10.4.1.4		
10.4.2	Level II Short Run Electrical Characteristics	Part 6, Sec. 10.4.2		
10.4.2.1	Level II Short Run Transmitter Characteristics	Part 6, Sec. 10.4.2.1		
10.4.2.1A	Level II short run transmitter must meet the output electrical specifications specified in Table 10-2 and detailed in Section 10.4.2.1.			
10.4.2.1B	Level II short run transmitter must meet the output jitter specifications specified in Table 10-3 and detailed in Section 10.4.2.1.			
10.4.2.2	Level II Short Run Receiver Characteristics	Part 6, Sec. 10.4.2.2		
10.4.2.2A	Level II short run receiver must meet the electrical input specifications specified in Table 10-6 and detailed in Section 10.4.2.2.			
10.4.2.2B	Level II short run receiver must meet the input jitter tolerance specifications specified in Table 10-7 and detailed in Section 10.4.2.2.			
10.4.2.3	Level II Short Run Link and Jitter Budgets	Part 6, Sec. 10.4.2.3		
10.4.2.3A	Level II short run link must meet the informative loss, skew and jitter budget specifications specified in Table 10-10.			
10.4.2.3B	Level II short run link must meet the high frequency jitter budget specifications specified in Table 10-11.			
10.5	Level II Long Run Interface General Requirements	Part 6, Sec. 10.5	Long run only	
10.5.1	Long Run Jitter and Inter-operability Methodology	Part 6, Sec. 10.5.1		
10.5.1.1	Level II long run link must meet the channel compliance specifications specified in section 10.5.1.1.	Part 6, Sec. 10.5.1.1		

10.5.1.2	Level II long run transmitter must meet the inter-operability requirements specified in Section 10.5.1.2.	Part 6, Sec. 10.5.1.2		
10.5.1.3	Level II long run receiver must meet the inter-operability requirements specified in Section 10.5.1.3.	Part 6, Sec. 10.5.1.3		
10.5.2	Level II Long Run Electrical Characteristics	Part 6, Sec. 10.5.2		
10.5.2.1	Level II Long Run Transmitter Characteristics	Part 6, Sec. 10.5.2.1		
10.5.2.1A	Level II long run transmitter must meet the output electrical specifications specified in Table 10-13 and detailed in Section 10.5.2.1.			
10.5.2.1B	Level II long run transmitter must meet the output jitter specifications specified in Table 10-14 and detailed in Section 10.5.2.1.			
10.5.2.2	Level II Long Run Receiver Characteristics	Part 6, Sec. 10.5.2.2		
10.5.2.2A	Level II long run receiver must meet the electrical input specifications specified in Table 10-17 and detailed in Section 10.5.2.2.			
10.5.2.2B	Level II long run receiver must meet the receiver jitter tolerance requirements specified in Section 10.5.2.2.8.			
10.5.3	Level II Long Run Link and Jitter Budgets	Part 6, Sec. 10.5.3		
10.5.3A	Level II long run link must meet the informative loss, skew and jitter budget specifications specified in Table 10-19.			
10.5.3B	Level II long run link must meet the high frequency jitter budget specifications specified in Table 10-20.			
10.6	Level II Medium Run Interface General Requirements	Part 6, Sec. 10.6	Medium run only	
10.6.1	Medium Run Jitter and Inter-operability Methodology	Part 6, Sec. 10.6.1		
10.6.1.1	Level II medium run link must meet the channel compliance specifications specified in section 10.6.1.1.	Part 6, Sec. 10.6.1.1		
10.6.1.2	Level II medium run transmitter must meet the inter-operability requirements specified in Section 10.6.1.2.	Part 6, Sec. 10.6.1.2		
10.6.1.3	Level II medium run receiver must meet the inter-operability requirements specified in Section 10.6.1.3.	Part 6, Sec. 10.6.1.3		
10.6.2	Level II Medium Run Interface Electrical Characteristics	Part 6, Sec. 10.6.2		
10.6.2.1	Level II Medium Run Transmitter Characteristics	Part 6, Sec. 10.6.2.1		
10.6.2.1A	Level II medium run transmitter must meet the output electrical specifications specified in Table 10-22 and detailed in Section 10.6.2.1.			
10.6.2.1B	Level II medium run transmitter must meet the output jitter specifications specified in Table 10-23 and detailed in Section 10.5.2.1.			
10.6.2.2	Level II Medium Run Receiver Characteristics	Part 6, Sec. 10.6.2.2		
10.6.2.2A	Level II medium run receiver must meet the electrical input specifications specified in Table 10-26 and detailed in Section 10.6.2.2.			
10.6.2.2B	Level II medium run receiver must meet the receiver jitter tolerance requirements specified in Section 10.6.2.2.8.			

10.6.3	Level II Medium Run Link and Jitter Budgets	Part 6, Sec. 10.6.3		
10.6.3A	Level II medium run link must meet the informative loss, skew and jitter budget specifications specified in Table 10-28.			
10.6.3B	Level II medium run link must meet the high frequency jitter budget specifications specified in Table 10-29.			