RapidIO Trade Association

Suite 325, 3925 W. Braker Lane
Austin, TX 78759
512-305-0070 Tel.
512-305-0009 FAX.

TWG Second Showing

Item 12-01-00000.001

Subject: Protocol enhancements for 3.0 Specifications

Background: There are a few protocol enhancements which should be made to improve the competitiveness of RapidIO. This showing proposes an increase in the maximum size of device IDs to 32 bits. This showing was split off from showing 11-11-00001.004 for inclusion in the 3.0 specification.

Contributor: Barry Wood, IDT

Comment Expiration Date:

Distribution: RapidIO TA Technical Working Group members

1.2 Discussion

While it is agreed to increase the maximum device ID size from 2 bytes to 4 bytes, the programming model for accessing the routing tables are more controversial. To facilitate completion of a consistent 3.0 specification, this showing has been split off from showing 11-10-00001.004 in order to include the impact of increasing the device ID size on the maximim packet size.

1.2.1 Discussion of Register Enhancements

Registers related to device ID occur in Part 3, Part 8, and Part 11.

The complexity of enhancing the programming model is as follows:

- Every status bit which indicates whether the current “large” (16 bit) deviceIDs are supported must have another status bit which indicates support for TT code 0b10.

- Every register field which indicates a deviceID must be extended to support the extended deviceID size.

The addition of status bits should not be too problematic. The addition of registers is, because generally the existing deviceID register fields consume an entire register so a new register must be created. The register map does not always have intuitively obvious locations to add registers.

1.3 Proposal

Changed or added text is indicated by underlining that text. Removed text is indicated by ~~striking the text~~.

Changes are proposed to Part 3, 6, and 8.

1.3.1 Part 3 Enhancements

Change section 2.4 of Part 3 of the specification as indicated by the underlined text below:

“2.4 Field Alignment and Definition

> The *RapidIO Part 3: Common Transport Specification* adds a transport type (tt) field to the logical specification packet that allows four different transport packet types to be specified. The tt field indicates which type of additional transport fields are added to the packet.
>
> The three fields (tt, destinationID, and sourceID) added to the logical packets allow for three different sizes of the device ID fields, a Dev32 (32 bit), Dev16

(16-bit), and a Dev8 (8-bit), as shown in Table 1-1. The three sizes of device ID fields allow three different system scalability points to optimize packet header overhead, and only affix additional transport field overhead if the additional addressing is required. The Dev32 enables large system scalability while allowing packets to be routed to specific functions within a device based on destination ID. The Dev8 fields allow a maximum of 256 devices to be attached to the fabric. The Dev16 fields allow systems with up to 65,536 devices. The Dev32 fields allow systems with up to 4,294,967,296 devices.

Table 1-1. tt Field Definition

tt	Definition
0b00	8-bit deviceID fields
0b01	16-bit deviceID fields
0b10	32-bit deviceID fields
0b11	Reserved

Figure 1-1 shows the transport header definition bit stream. The shaded fields are the bits associated with the logical packet definition that are related to the transport bits. Specifically, the field labeled “Logical ftype” is the format type field defined in the logical specifications. This field comprises the first four bits of the logical packet. The second logical field shown (“Remainder of logical packet”) is the remainder of the logical packet of a size determined by the logical specifications, not including the logical ftype field which has already been included in the combined bit stream. The unshaded fields (tt, destinationID and sourceID fields) are the transport fields added to the logical packet by the common transport specification.

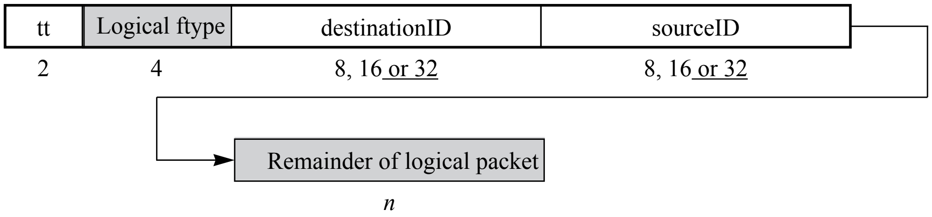

Figure 1-1. Destination-Source Transport Bit Stream

“

Change diagram 2-4 as indicated by the underlined text below:

“

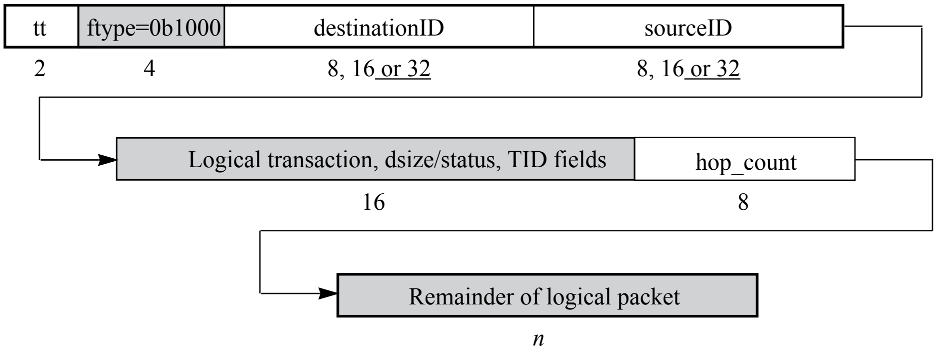
tt
ftype=0b1000
destinationID
sourceID
2
4
8, 16 or 32
8, 16 or 32
Logical transaction, dsize/status, TID fields
hop_count
16
8
Remainder of logical packet
n

Figure 1-2. Maintenance Packet Transport Bit Stream

“

Change the Part 3 Chapter 2 Common Transport Registers summary table as identified by the underlined text below.

“

Table 1-2. Common Transport Register Map

Configuration Space Byte Offset	Register Name
0x0-C	Reserved
0x10	Processing Element Features CAR
0x14–30	Reserved
0x34	Switch Route Table Destination ID Limit CAR
0x38-5C	Reserved
0x60	Base Device ID CSR
0x64	Dev32 Base Device ID CSR
0x68	Host Base Device ID Lock CSR
0x6C	Component Tag CSR
0x70	Standard Route Configuration Destination ID Select CSR
0x74	Standard Route Configuration Port Select CSR
0x78	Standard Route Default Port CSR
0x7C–FC	Reserved
0x100–FFFC	Extended Features Space
0x10000–FFFFFC	Implementation-defined Space

“

Change the Part 3 documentation of the Processing Element Features CAR by adding an indication for Dev32 support, as indicated below.

“

1.3.2 Processing Element Features CAR (Configuration Space Offset 0x10)

The processing element features CAR identifies the major functionality provided by the processing element. The bit settings are shown in Table 1-3.

Table 1-3. Bit Settings for Processing Element Features CAR

Bits	Name	Description
0–19	—	Reserved
20	Dev32 Support	0b0 - PE does not support Common Transport 32 bit device IDs 0b1 - PE supports Common Transport 32 bit device IDs
21	—	Reserved
22	Extended route table configuration support	0b0 - Switch PE does not support the extended route table configuration mechanism 0b1 - Switch PE supports the extended route table configuration mechanism (can only be set if bit 23 is set and bit 20 is clear)
23	Standard route table configuration support	0b0 - Switch PE does not support the standard route table configuration mechanism 0b1 - Switch PE supports the standard route table configuration mechanism
24–26	—	Reserved
27	Common transport large system support	0b0 - PE does not support common transport large systems 0b1 - PE supports common transport large systems
28–31	—	Reserved

“

Add a Base Device ID CSR for Dev32 to Part 3, as indicated below:

“

1.3.3 Dev32 Base Device ID CSR (Configuration Space Offset 0x64)

The Dev32 Base Device ID CSR contains the 32 bit Base Device ID value for the processing element. A device may have multiple device ID values, but these are not defined in a standard CSR. The bit settings are shown in Table 2-5.

Table 1-4. Bit Settings for Dev32 Base Device ID CSR

Bits	Name	Reset Value	Description
0-31	Dev32_Base_DeviceID	see footnote[1]	This is the 32 bit base device ID of the device in a Dev32 transport system (end point devices only)

[1] TheDev32_Base_DeviceID reset value is implementation dependent

“

1.3.4 Part 6 Enhancements

Change section Part 6 Section 2.5 Maximum Packet Size as indicated by the underlined text:

“

1.4 Maximum Packet Size

The RapidIO Specification does not contain an overall specification for the maximum size of a packet that a logical layer may pass to the transport layer or the transport layer may pass to a physical layer. Maximum sizes can only be determined by examining the format of each packet type at the logical layer and the operation of the transport and physical layers.

The longest packets are those containing an operand address within the destination device, an operand size and a maximum length payload (256 bytes). Currently the largest packet format is the type 5 (write class) format, defined in the I/O Logical specification. The sizes of the components of the maximum packet are shown in more detail in Table 1-6.

Table 1-5. Maximum Packet Size, Baud Rate Class 1 and 2

Field	Size (bytes)	Layer	Notes
Header	2	Logical, Transport, Physical	See Figure 2-2
Source ID	4	Transport	Dev32
Destination ID	4	Transport	Dev32
Trans/wrsize	1	Logical	Type 5 (write class)
srcTID	1	Logical	Type 5 (write class)
Address	8	Logical	Type 5 (write class); includes Extended_address, Address, Wdptr, and Xambs
Payload	256	Logical	Maximum data payload
CRC	4	Physical	Includes two CRC additional bytes for packets greater than 80 bytes.
Total	280		

Table 1-6. Maximum Packet Size, Baud Rate Class 3

Field	Size (bytes)	Layer	Notes
Header	2	Logical, Transport, Physical	See Figure 2-2
Source ID	4	Transport	Dev32
Destination ID	4	Transport	Dev32
Trans/wrsize	1	Logical	Type 5 (write class)
srcTID	1	Logical	Type 5 (write class)
Address	8	Logical	Type 5 (write class); includes Extended_address, Address, Wdptr, and Xambs
Payload	256	Logical	Maximum data payload
CRC	8	Physical	Includes two CRC additional bytes for packets greater than 80 bytes, and one more 4 byte link-level CRC for Baud Rate Class 3 Devices.
Total	284		

The maximum transmitted packet size permitted by the LP-Serial specification shall be 280 bytes for Baud Rate Class 1 and 2 links, and 284 bytes for Baud Rate Class 3 links. This includes all packet logical, transport, and physical layer header information, data payload, and required CRC bytes, but does not include any packet delimiting control symbols or other necessary physical layer control information.

“

1.4.1 Part 8 Enhancements

Several registers and bit fields are added to the Part 8 Error Management/Hot Swap Extensions registers section. Changes to the register map are indicated with change bars in the updated table below:

“

Table 1-7. Error Management/Hot Swap Extensions Register Requirements

	Block Byte Offset	Register Name	Error Mgmt Only	Hot Swap & Error Mgmt	Hot Swap Only
General	0x0	Error Management/Hot Swap Extensions Block Header	X	X	X
	0x4	Error Management/Hot Swap Extensions Block CAR	X	X	X
	0x8	Logical/Transport Layer Error Detect CSR	X	X	-
	0xC	Logical/Transport Layer Error Enable CSR	X	X	-
	0x10	Logical/Transport Layer High Address Capture CSR	X	X	-
	0x14	Logical/Transport Layer Address Capture CSR	X	X	-
	0x18	Logical/Transport Layer Device ID Capture CSR	X	X	-
	0x1C	Logical/Transport Layer Control Capture CSR	X	X	-
	0x20	Logical/Transport Layer Dev32 Destination ID Capture CSR	X	X	-
	0x24	Logical/Transport Layer Dev32 Source ID Capture CSR	X	X	-
	0x28	Port-write Target deviceID CSR	X	X	X
	0x2C	Packet Time-to-live CSR	X	X	-
	0x30	Port-write Dev32 Target deviceID CSR	X	X	-
	0x34-3C	Reserved			

Table 1-7. Error Management/Hot Swap Extensions Register Requirements

	Block Byte Offset	Register Name	Error Mgmt Only	Hot Swap & Error Mgmt	Hot Swap Only
Port 0	0x40	Port 0 Error Detect CSR	X	X	X
	0x44	Port 0 Error Rate Enable CSR	X	X	X
	0x48	Port 0 Attributes Capture CSR	X	X	-
	0x4C	Port 0 Packet/Control Symbol Capture 0 CSR	X	X	-
	0x50	Port 0 Packet Capture 1 CSR	X	X	-
	0x54	Port 0 Packet Capture 2 CSR	X	X	-
	0x58	Port 0 Packet Capture 3 CSR	X	X	-
	0x5C	Port 0 Packet Capture 4 CSR	X	X	-
	0x60-64	Reserved			
	0x68	Port 0 Error Rate CSR	X	X	-
	0x6C	Port 0 Error Rate Threshold CSR	X	X	-
	0x70	Port 0 Link Uninit Discard Timer CSR	-	X	X
	0x74-7C	Reserved			
Port 1	0x80	Port 1 Error Detect CSR	X	X	X
	0x84	Port 1 Error Rate Enable CSR	X	X	X
	0x88	Port 1 Attributes Capture CSR	X	X	-
	0x8C	Port 1 Packet/Control Symbol Capture 0 CSR	X	X	-
	0x90	Port 1 Packet Capture 1 CSR	X	X	-
	0x94	Port 1 Packet Capture 2 CSR	X	X	-
	0x98	Port 1 Packet Capture 3 CSR	X	X	-
	0x9C	Port 1 Packet Capture 4 CSR	X	X	-
	0xA0-A4	Reserved			
	0xA8	Port 1 Error Rate CSR	X	X	-
	0xAC	Port 1 Error Rate Threshold CSR	X	X	-
	0xB0	Port 1 Link Uninit Discard Timer CSR	-	X	X
	0xB4-BC	Reserved			
Ports 2-14	0xC0–3FC	Assigned to Port 2-14 CSRs Register implementation requirements are the same as for port 0			

Table 1-7. Error Management/Hot Swap Extensions Register Requirements

	Block Byte Offset	Register Name	Error Mgmt Only	Hot Swap & Error Mgmt	Hot Swap Only
Port 15	0x400	Port 15 Error Detect CSR	X	X	X
	0x404	Port 15 Error Rate Enable CSR	X	X	X
	0x408	Port 15 Attributes Capture CSR	X	X	-
	0x40C	Port 15 Packet/Control Symbol Capture 0 CSR	X	X	-
	0x410	Port 15 Packet Capture 1 CSR	X	X	-
	0x414	Port 15 Packet Capture 2 CSR	X	X	-
	0x418	Port 15 Packet Capture 3 CSR	X	X	-
	0x41C	Port 15 Packet Capture 4 CSR	X	X	-
	0x420-424	Reserved			
	0x428	Port 15 Error Rate CSR	X	X	-
	0x42C	Port 15 Error Rate Threshold CSR	X	X	-
	0x430	Port 15 Link Uninit Discard Timer CSR	-	X	X
	0x434-43C	Reserved			

“

Change the definition of Port-write Target deviceID CSR as indicated by the underlined text:

“

1.4.1.1 Port-write Target deviceID CSR (Block Offset 0x28)

This register contains the target deviceID to be used when a device generates a Maintenance port-write operation to report errors to a system host. The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-8. Unless otherwise specified, the bits and bit fields in this register are read/write.

Table 1-8. Bit Settings for Port-write Target deviceID CSR

Bit	Name	Reset Value	Description
0-7	deviceID_msb	0x00	This is the most significant byte of the port-write target deviceID (large transport systems only)
8-15	deviceID	0x00	This is the port-write target deviceID

Table 1-8. Bit Settings for Port-write Target deviceID CSR

Bit	Name	Reset Value	Description
16	Dev8_or_16	0b0	Dev8 or Dev16 deviceID size to use for a port-write 0b0 - use the small transport deviceID 0b1 - use the large transport deviceID This bit field controls the deviceID size to use for a port-write when Dev32_PW is 0.
17	Dev32_PW	0b0	This bit field shall be implemented for devices which have bit 20 set in the Processing Element Features CAR. This bit field controls the use of Dev32 deviceID size for a port-write 0b0 - port-write deviceID size is controlled by Dev8_or_16 bit 0b1 - use Dev32 deviceID size for port-write
18-31	—		Reserved

“

New register definitions for Part 8 are captured below.

“

1.4.1.2 Logical/Transport Layer Dev32 Destination ID Capture CSR (Block Offset 0x20)

This register contains error information. It is locked when an error is detected and the corresponding enable bit is set. This register shall be implemented for devices which have bit 20 set in the Processing Element Features CAR. The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-9. This register shall be read only.

Table 1-9. Bit Settings for Logical/Transport Layer Dev32 Destination ID Capture CSR

Bit	Name	Reset Value	Description
0-31	Dev32 DestID	All 0’s	The Dev32 destination ID associated with the error.

1.4.1.3 Logical/Transport Layer Dev32 Source ID Capture CSR (Block Offset 0x24)

This register contains error information. It is locked when an error is detected and the corresponding enable bit is set. This register shall be implemented for devices which have bit 20 set in the Processing Element Features CAR. The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-10. This register shall be read only.

Table 1-10. Bit Settings for Logical/Transport Layer Dev32 Source ID Capture CSR

Bit	Name	Reset Value	Description
0-31	Dev32 SrcID	All 0’s	The Dev32 source ID associated with the error.

1.4.1.4 Port-write Dev32 Target DeviceID CSR (Block Offset 0x30)

This register contains the Dev32 target deviceID to be used when a device generates a Maintenance port-write operation to report errors to a system host and the Dev32_PW bit is set. This register shall be implemented for devices which have bit 20 set in the Processing Element Features CAR. The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-11. Unless otherwise specified, the bits and bit fields in this register are read/write.

Table 1-11. Bit Settings for Port-write Dev32 Target DeviceID CSR

Bit	Name	Reset Value	Description
0-31	Dev32 DestID	All 0's	The port-write Dev32 target device ID.

1.4.1.5 Port *n* Packet Capture 4 CSR (Block Offset 0x5C, 9C,..., 41C)

The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-12. This register shall be implemented for devices which have bit 20 set in the Processing Element Features CAR. The contents of the register are valid and locked when the Capture Valid info bit of the Port n Attributes Capture SCR is set. Unless otherwise specified, the bits and bit fields in this register are read/write.

Table 1-12. Bit Settings for Port *n* Packet Capture 4 CSR

Bit	Name	Reset Value	Description
0-31	Capture 4	All 0s	If the info_type field of the Port n Attributes Capture CSR is "packet", packet Bytes 16-19. Otherwise, implementation specific.

“