# RapidIO Trade Association

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

TWG Second Showing
Item 10-06-00000.010

Subject: Hot Swap Specification

Background: RapidIO requires an interoperable hot swap spec. This document proposes mechanisms which remedy some shortcomings with respect to hot swap in the Gen 1 and Gen 2 specs.

Contributor: Barry Wood, IDT

Comment Expiration Date:

Distribution: RapidIO TA Technical Working Group members

## 1.2 Introduction

The RapidIO specification does allow Field Replaceable Units (FRUs) to be removed and inserted under system control. However, doing so requires implementation specific knowledge of the boards/components on the FRU. If an FRU is being replaced with a different type of FRU, it is difficult to establish communication.

## 1.3 Hot Swap Use Cases

The requirements for hot swap, and a discussion of the deficiencies in the Gen 2 specification, are found in “RapidIO 3.0 Hot Swap Straw Man” in the ‘Presentations’ folder of the TWG group.

The sections below capture these use cases. Note that the existence of a central maintenance software entity responsible for bringing FRUs into and out of the system is not a part of all scenarios. It should be possible for FRUs to remove themselves gracefully from the system. It should also be possible for FRUs to be removed from a system without warning, and inserted into the system under their own control.

### 1.3.1 Controlled Removal

In this case, the FRU about to be removed can coordinate it’s removal with the rest of the system. The following steps occur:

1. Traffic being sent to the FRU is brought to a graceful halt. This can be done by a variety of means, either through software communication or hardware isolation mechanism.”Graceful” means that all outstanding transactions are removed from the system without a reported error.
2. Traffic sent from the FRU is brought to a graceful halt. Again, this can be done by a variety of means, either through software communication or hardware isolation mechanism. “Graceful” means that all outstanding transactions are removed from the system without a reported error.
3. The FRU is removed.

### 1.3.2 Uncontrolled Removal

In this case, the FRU is removed without coordination with the rest of the system.

Isolation/packet discard mechanisms whose purpose is to clean up outstanding transactions in the event of failure must be configured before uncontrolled removal can be successful. Depending on the system, this can include Time To Live as well as other Error Management mechanisms for packet discard.

### 1.3.3 FRU Insertion

In this case, an FRU is inserted into a running system. The following steps must occur:

1. The rest of the system must be configured to allow traffic to be sent to, and responses received from, the newly inserted FRU.
2. The rest of the system must be configured to allow the FRU to send traffic and receive responses.
3. The FRU begins packet transmission and reception.

### 1.3.4 Aborted Controlled Removal

This case is the same as Controlled Removal, with the exception that the board is not actually removed from the system. Instead, the Insertion process is performed on a board which was already in the system. This implies that all links are still operational.

## 1.4 Discussion

There are three areas where the Gen 2 specifications are deficient when it comes to hot swap. They are discussed in detail in the following sections.

### 1.4.1 Packet Discard for Hot Swap

RapidIO's reliable delivery of packets simplifies system design. RapidIO has configurable mechanisms to discard packets when the target processing element cannot receive them. The standard Error Management Extensions "leaky bucket" mechanism can be used to discard packets when a link partner has disappeared. Practically speaking, the leaky bucket thresholds for a hot swap event must be set so low that they would result in link failure for single, recoverable events.

The implication is that a specific mechanism to discard packets in the event that a link partner has disappeared is required.

This mechanism is required for switches which support hot swap. The mechanism may also be necessary for endpoints which have multiple RapidIO ports and packet forwarding capability.

The mechanism for detecting that a link partner has disappeared could be a timer. There are a number of options for this timers operation:

- Start timer once silence is received from the link partner, indicating that the link partner has either been removed or has started the link initialization sequence. The timer runs until PORT_OK is set. If a programmable threshold time is met, start packet discard.
- Start timer after this ends silence timer has expired. The timer runs while in SEEK state. If a programmable threshold time is met, start packet discard.

If a timer approach is proposed, the timer specification must meet two requirements. It must be capable of operating quickly enough to minimize congestion if a link reinitializes in a system which cannot tolerate lengthy delays. In terms of packets, the timer should support periods as short as the time it takes to transmit 1 to 10 packets. It must also be capable of operating slowly enough to be certain that a link partner has either failed to train, or has been removed. For example, the timer should support a long enough period so that it does not fire if a cable or board is accidentally unplugged and then immediately replaced. This certainty requires the timer to support periods which are more than 1 second. Link initialization periods are trending longer as well. For example, the 802.3 10kr standard requires 0.5 seconds for each lane initialization attempt. As SerDes speeds increase, the time required to optimize SerDes equalization parameters increases as well.

Another option is to have a configuration bit (DISCARD_FOR_PORT_UNINIT?) which controls packet discard when a port does not have PORT_OK asserted. Packet discard is triggered when a port transitions from PORT_OK to PORT_UNINIT.

Yet another option, which leverages current functionality, is to set the PORT_LOCKOUT bit when a port transitions from PORT_OK to PORT_UNINIT if DISCARD_FOR_PORT_UNINIT is set. Packets sent to this port from within the device are discarded if the “Drop Packet Enable” bit is set.

### 1.4.2 Link Recovery for Link Partner Insertion

Link recovery for link partner insertion is discussed and a solution proposed in showing 11-01-00001.

### 1.4.3 Hot Swap Link State Notification

When a link partner is removed, or when a link partner reappears, a standard notification mechanism should exist to let system software know of the event.

Interrupts are implementation specific, and beyond the scope of this showing.

The Error Management Extensions port-write could be modified to carry information about link partner appearances and disappearances.

In order to allow the port-writes to be forward/backwards compatible, the status bits for a hot swap event should be kept within the same error status registers which are part of the port-write now.

Given that we don’t want hot-swap support to require implementation of the Error Management spec, it’s proposed that

- “Link Partner Trained” and “Link Partner Removed” events should be added to the Error Management Extensions Port n Error Detect CSR.
    - — Link Partner Trained indicates that a port’s link_initialized variable has transitioned from deasserted to asserted.

    - Link Partner Removed indicates a port’s link_initialized variable has transitioned from asserted to deasserted.
    - These events do not contribute to the Error Rate Counter
    - These events, if enabled, trigger transmission of a port-write/assertion of an interrupt when detected.
- Corresponding ‘Enable’ bits should be added to the Error Management Extensions Port n Error Rate Enable CSR.
- A new version of the Error Management Registers Block should be defined which includes the new events/bits.
- A separate “Hot Swap Block” should be defined which includes
    - “Link Partner Trained” and “Link Partner Removed” bits in the Port n Error Detect/Port n Error Rate Enable CSR
    - Port Write control register, as per definition in Part 8 Error Management Extensions

## 1.5 Proposed Changes to Part 8 Error Management Extensions

The proposed design extends the existing Error Management Extensions specification to incorporate hot swap functions which can be implemented by themselves, incorporated into Error Management registers, or not included.

### 1.5.1 Change Name of Part 8

Change the name of Part 8 from “Error Management Extensions Specification” to “Error Management/Hot Swap Extensions Specification”. Note that this applies to the title on the cover page, as well as the document master page layout.

### 1.5.2 Additional “Hot Swap Extensions” Description

Add the following text to the end of section 1.2 Physical Layer Extensions:

“The Hot Swap Extensions consist of the following registers and register bit extensions, which allow software to be notified of the addition and removal of processing elements:

- Port Write Target ID CSR defined in section 2.3.2.8
- (Extensions to the) Port n Error Detect CSR defined in Section 2.3.2.10
- (Extensions to the) Port n Error Rate Enable CSR defined in Section 2.3.2.11
- Port n Link Uninit Discard Timer CSR defined in Section 2.3.2.21”

Add the following new section:

“1.2.6 Hot Swap Extensions

When a Field Replaceable Unit (FRU) is inserted into a running system, it may be necessary to immediately inform system software. Similarly, when an FRU is removed from a running system, it may be necessary to immediately inform system software. The Link Uninit to OK Transition event can be used to inform system software of the insertion of an FRU. The Link OK to Uninit Transition event can be used to inform system software of the removal of an FRU.

In the event that an FRU is removed from a system unexpectedly, the number of physical layer errors detected is uncertain. It is not possible to set the Physical Layer Error Management extensions thresholds, as described in sections 1.2.2 and 1.2.3, to differentiate between an expected bit error rate and FRU removal. The Hot Swap Extensions uses a timeout period for link reinitialization, the Port n Link Uninit Discard Timer CSR, to detect when a link has been unavailable for a period of time deemed excessive by the system. When the Port n Link Uninit Discard Timer period expires, packets are discarded to avoid system congestion. Depending on the system design, the congestion could prevent system software from handling the unexpected FRU removal, which could lead to system failure.

A port-write may be sent to inform system software of a Hot Swap Extensions event. The Hot Swap Extensions events are incorporated into the Port n Error Detect CSR, as the contents of this CSR is sent a port-write. The Hot Swap Extensions events are also included in the Port n Error Rate Enable CSR, as this is the standard register which controls notification and information capture for physical layer events.

However, unlike the Error Management Extensions physical layer events, the removal or insertion of an FRU is not a correctable error. For this reason, unlike Error Management Extensions events, Hot Swap Extension events do not contribute to the error reporting thresholds described in section 1.2.2/1.2.3, do not cause any error information to be latched and do not cause the Port *n* Packet/Control Symbol Capture 0-3 CSRs to lock.”

### 1.5.3 New Register Map Section

Add the following content after the first paragraph of Part 8, section 2.3.1 Register Map:

“The registers which appear in the Error Management/Hot Swap Extensions Register Block vary based on the functionality indicated in the Error Management/Hot Swap Extensions Block CAR. Table 1-1, on page -9 describes what registers shall be implemented based on the value of the Error Management/Hot Swap Extensions Block CAR. The register offsets and names are listed, along with three columns that indicate which registers must be implemented. An “X” in the column means that the register shall be implemented.

### Table 1-1. 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 | - |
| | 0x20-24 | Reserved | | | |
| | 0x28 | Port-write Target deviceID CSR | X | X | X |
| | 0x2C | Packet Time-to-live CSR | X | X | - |
| | 0x30-3C | Reserved | | | |
| 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-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 | | | |

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

| | Block Byte Offset | Register Name | Error Mgmt Only | Hot Swap & Error Mgmt | Hot Swap Only |
|---|---|---|---|---|---|
| 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-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<br>Register implementation requirements are the same as for port 0 | | | |
| 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-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 | | | |

## 1.5.4 New “Error Management/Hot Swap Extensions Block CAR”

Create a new register, the “Error Management/Hot Swap Extensions Block CAR”, as follows:

### 1.5.4.1 Error Management/Hot Swap Extension Block CAR (Block Offset 0x4)

This register indicates the supported Error Management Extension and Hot Swap Extension functionality. The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-2. The register is read-only.

**Table 1-2. Bit Settings for Error Management/Hot Swap Extension Block CAR**

| Bit | Name | Reset Value | Description |
|---|---|---|---|
| 0 | Error Management Extensions Not Implemented | Implementation Specific | Indicates whether or not Error Management Extensions functionality (and registers) have been implemented.<br>0b0 - all registers and bit fields specific to Error Management Extensions shall be implemented.<br>0b1 - all registers and/or bit fields specific to Error Management Extensions shall not be implemented. |
| 1 | Hot Swap Extensions Implemented | Implementation Specific | Indicates whether or not Hot Swap functionality and registers have been implemented.<br>0b0 - all registers and bit fields specific to Hot Swap Extensions support shall not be implemented.<br>0b1 - all registers and bit fields specific to Hot Swap Extensions support shall be implemented. |
| 2-31 | — | | Reserved |

## 1.5.5 Modification of Port n Error Detect CSR

Change the definition of the Port n Error Detect CSR by adding the Link OK to Uninit Transition, Link Uninit Packet Discard Active, and Link Uninit to OK Transition bits. Also add text to the register description, shown underlined below.

### 1.5.5.1 Port *n* Error Detect CSR (Block Offset 0x40, 80,..., 400)

The Port *n* Error Detect Register indicates the physical layer errors that have been detected by the Port *n* hardware since the register was last cleared. The register is cleared by software writing the register with the data 0x0000_0000.

The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-3. Unless otherwise specified, the bits and bit fields in this register are read/write.

The 2 columns at the right of Table 1-3 indicate which bit fields must be implemented for Error Management Extensions, and for Hot Swap Extensions. An “X” in these columns means that the bit shall be implemented for that extension.

**Table 1-3. Bit Settings for Port *n* Error Detect CSR**

| Bit | Name | Reset Value | Description | Error Mgmt | Hot Swap |
|---|---|---|---|---|---|
| 0 | Implementation specific error | 0b0 | An implementation specific error has been detected | X | - |
| 1 | Link OK to Uninit Transition | 0b0 | The link has transitioned from a link initialized to link uninitialized state. | - | X |
| 2 | Link Uninit Packet Discard Active | 0b0 | The Link Uninit Discard Timer CSR period has expired. | - | X |
| 3 | Link Uninit to OK Transition | 0b0 | The link has transitioned from a link uninitialized to link initialized state. | - | X |
| 4-7 | — | | Reserved | | |
| 8 | Received S-bit error | 0b0 | Received a packet/control symbol with an S-bit parity error (parallel) | X | - |
| 9 | Received corrupt control symbol | 0b0 | Received a control symbol with a bad CRC value (serial)<br>or<br>Received a control symbol with a true/complement mismatch (parallel) | X | - |
| 10 | Received acknowledge control symbol with unexpected ackID | 0b0 | Received a packet-accepted or packet-retry control symbol with an unexpected ackID | X | - |
| 11 | Received packet-not-accepted control symbol | 0b0 | Received packet-not-accepted control symbol | X | - |
| 12 | Received packet with unexpected ackID | 0b0 | Received packet with unexpected ackID value - out-of-sequence ackID | X | - |
| 13 | Received packet with bad CRC | 0b0 | Received packet with a bad CRC value | X | - |
| 14 | Received packet exceeds 276 Bytes | 0b0 | Received packet which exceeds the maximum allowed size | X | - |

Table 1-3. Bit Settings for Port *n* Error Detect CSR

| Bit | Name | Reset Value | Description | Error Mgmt | Hot Swap |
|---|---|---|---|---|---|
| 15 | Received illegal or invalid character | 0b0 | Received an 8B/10B code-group that is invalid (a code-group that does not have a 8B/10B decode given the current running disparity) or illegal (a code-group that is valid, but whose use is not allowed by the LP-Serial protocol). This bit may be set in conjunction with bit 29 Delineation error. The implementation of this bit is optional, but strongly recommended. (serial) | X | - |
| 16 | Received data character in IDLE1 sequence | 0b0 | Received a data character in an IDLE1 sequence. This bit may be set in conjunction with bit 29 Delineation error. The implementation of this bit is optional, but strongly recommended. (serial) | X | - |
| 17 | Loss of descrambler synchronization | 0b0 | Loss of receiver descrambler synchronization while receiving scrambled control symbol and packet data. This bit shall be implemented only if port n supports descrambling of packet and control symbol data. (serial) | X | - |
| 18-25 | — | | Reserved | | |
| 26 | Non-outstanding ackID | 0b0 | When there are outstanding ackIDs, a link_response was received with an ackID that is not outstanding | X | - |
| 27 | Protocol error | 0b0 | An unexpected control symbol was received | X | - |
| 28 | Frame toggle edge error | 0b0 | FRAME signal toggled on falling edge of receive clock (parallel) | X | - |
| 29 | Delineation error | 0b0 | Received an 8B/10B code-group that is invalid (a code-group that does not have a 8B/10B decode given the current running disparity), illegal (a code-group that is valid, but whose use is not allowed by the LP-Serial protocol) or that is in a position in the received code-group stream that is not allowed by the LP-Serial protocol (serial)<br>or<br>FRAME signal toggled on non-32-bit boundary (parallel) | X | - |
| 30 | Unsolicited acknowledgement control symbol | 0b0 | An unsolicited packet acknowledgement control symbol was received | X | - |
| 31 | Link timeout | 0b0 | A packet acknowledgement or link-response control symbol was not received within the specified timeout interval | X | - |

## Modification of Port n Error Rate Enable CSR

Change the definition of the Port n Error Rate Enable CSR by adding the Link OK to Uninit Transition Enable, Link Uninit Packet Discard Active Enable, and Link Uninit to OK Transition Enable bits. Also add text to the register description, shown underlined below.

### 1.5.5.2 Port *n* Error Rate Enable CSR (Block Offset 0x44, 84,..., 404)

This register contains Error Management Extensions bits that when set cause specific detected errors to increment the error rate counter in the Port *n* Error Rate Threshold Register and capture information about the error in and then lock the Port *n* Packet/Control Symbol Capture 0-3 CSRs. Without exception, bit “b” of this register controls the capture and counting of the Error Management Extensions detected error whose occurrence is indicated by bit “b” of the Port *n* Error Detect CSR.

This register also contain Hot Swap Extensions bits which control immediate notification for the event..There is no error information associated with the Hot Swap Extensions events. The Port *n* Packet/Control Symbol Capture 0-3 CSRs are not locked by the Hot Swap Extensions events. Without exception, bit “b” of this register controls the immediate notification of the Hot Swap Extensions detected event whose occurrence is indicated by bit “b” of the Port *n* Error Detect CSR

The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-4. Unless otherwise specified, the bits and bit fields in this register are read/write.

The 2 columns at the right of Table 1-3 indicate which bit fields must be implemented for Error Management Extensions, and for Hot Swap Extensions. An “X” in these columns means that the bit shall be implemented for that extension.

**Table 1-4. Bit Settings for Port *n* Error Rate Enable CSR**

| Bit | Name | Reset Value | Description | Error Mgmt | Hot Swap |
|---|---|---|---|---|---|
| 0 | Implementation specific error enable | 0b0 | Enable error rate counting of implementation specific errors | X | - |
| 1 | Link OK to Uninit Transition Enable | 0b0 | Enable event notification for Link OK to Uninit Transition events. | - | X |
| 2 | Link Uninit Packet Discard Active Enable | 0b0 | Enable event notification for Link Uninit Packet Discard Timer Active events. | - | X |
| 3 | Link Uninit to OK Transition Enable | 0b0 | Enable event notification for Link Uninit to OK Transition events. | - | X |
| 4-7 | — |  | Reserved |  |  |
| 8 | Received S-bit error enable | 0b0 | Enable error rate counting of a packet/control symbol with an S-bit parity error (parallel) | X | - |
| 9 | Received corrupt control symbol enable | 0b0 | Enable error rate counting of a corrupt control symbol | X | - |

## Table 1-4. Bit Settings for Port *n* Error Rate Enable CSR

| Bit | Name | Reset Value | Description | Error Mgmt | Hot Swap |
|---|---|---|---|---|---|
| 10 | Received out-of-sequence acknowledgement control symbol enable | 0b0 | Enable error rate counting of a packet-accepted or packet-retry control symbol with an unexpected ackID | X | - |
| 11 | Received packet-not-accepted control symbol enable | 0b0 | Enable error rate counting of received packet-not-accepted control symbols. | X | - |
| 12 | Received packet with unexpected ackID enable | 0b0 | Enable error rate counting of packet with unexpected ackID value - out-of-sequence ackID | X | - |
| 13 | Received packet with bad CRC enable | 0b0 | Enable error rate counting of packet with a bad CRC value | X | - |
| 14 | Received packet exceeds 276 Bytes enable | 0b0 | Enable error rate counting of packet which exceeds the maximum allowed size | X | - |
| 15 | Received illegal or invalid character enable | 0b0 | Enable error rate counting of reception of an 8B/10B code-group that is invalid or illegal. This bit shall be implemented only if bit 15 “Received illegal or invalid character” of the Port *n* Error Detect CSR is implemented. (serial) | X | - |
| 16 | Received data character in an IDLE1 sequence enable | 0b0 | Enable error rate counting of reception of a data character in an IDLE1 sequence. This bit shall be implemented only if bit 16 “Received data character in IDLE1 sequencer” of the Port n Error Detect CSR is implemented. (serial) | X | - |
| 17 | Loss of descrambler synchronization enable | 0b0 | Enable error rate counting of loss of receiver descrambler synchronization when scrambled control symbol and packet data is being received. This bit shall be implemented only if bit 17 “Loss of descrambler synchronization” of the Port n Error Detect CSR is implemented. (serial) | X | - |
| 18-25 | — |  | Reserved |  |  |
| 26 | Non-outstanding ackID enable | 0b0 | Enable error rate counting of link-responses received with an ackID that is not outstanding when there are outstanding ackIDs | X | - |
| 27 | Protocol error enable | 0b0 | Enable error rate counting of received unexpected control symbol symbols | X | - |
| 28 | Frame toggle edge error enable | 0b0 | Enable error rate counting of frame toggle edge errors (parallel) | X | - |
| 29 | Delineation error enable | 0b0 | Enable error rate counting of<br>FRAME signal toggled on non-32-bit boundary (parallel)<br>or<br>Reception of an 8B/10B code-group that is invalid, illegal or that is in a position in the received code-group stream that is not allowed by the LP-Serial protocol (serial) | X | - |

Table 1-4. Bit Settings for Port *n* Error Rate Enable CSR

| Bit | Name | Reset Value | Description | Error Mgmt | Hot Swap |
|---|---|---|---|---|---|
| 30 | Unsolicited acknowledgement control symbol enable | 0b0 | Enable error rate counting of received unsolicited packet acknowledgement control symbols | X | - |
| 31 | Link timeout enable | 0b0 | Enable error rate counting of link timeout errors | X | - |

## 1.5.6 Addition of Port n Link Uninit Discard Timer CSR

Create a new register, the “Port n Link Uninit Discard Timer CSR”, as follows:

### 1.5.6.1 Port n Link Uninit Discard Timer CSR (Block Offset 0x70, 0xB0, ..., 0x430)

The maximum value of the Link Uninit Timeout variable (0xFFFFFF) shall correspond to 6 to 12 seconds. The resolution of the Link Uninit Timeout variable shall be (maximum Link Uninit Timeout interval)/($2^{24}$-1). The use and meaning of the bits and bit fields of this register shall be as specified in Table 1-5. Unless otherwise specified, the bits and bit fields in this register shall be readable and writable.

Table 1-5. Bit Settings for Port n Link Uninit Discard Timer CSR

| Bit | Name | Reset Value | Description |
|---|---|---|---|
| 0-23 | Link Uninit Timeout | 0x000000 | This timer shall start counting when port_initialized transitions from asserted to deasserted. When this timer expires, all packets directed to this port from inside the device shall be discarded, and a “Link Uninit Packet Discard Active” event shall be detected.<br>Packet discard shall occur until the “Link Uninit Packet Discard Active” status bit is cleared.<br>The Link Uninit Discard Timer shall be disabled when Link Uninit Timeout is 0. |
| 24-31 | — |  | Reserved |