Raymond Chan, Sr. Applications Engineer, PLX Technology

1. Introduction

The PEX 8619 provides two mechanisms for error injection:

- PCI Express Advanced Error Reporting Status register bits, and Slot Control Status registers bits (which normally cannot be Set by software) can be made writable, to allow software to generate error Messages and Hot Plug interrupts, respectively, to enable testing of software response to errors and interrupts.
- ECC RAM can be intentionally injected with 1-bit errors and 2-bit errors. 1-bit errors are corrected by the ECC logic, and are not propagated out the egress port link.

2. Software Generation of Advanced Error Reporting Messages

When the Software Force Error Enable bit (Ports 0 register 1C8h[2]) is set in the switch, the Uncorrectable Error and Correctable Error Status registers (FB8h and FC4h, respectively), in each Port of the corresponding Station, are changed from RW1C to Read/Write (on alternate cycles). Then when software writes "1" to one or more of the implemented bits that are not already Set in the Uncorrectable Error and Correctable Error registers, the switch port will generate the corresponding Message (provided the corresponding Mask bit is not Set, and Messages are enabled in the Device Control register (70h)). Writing the value "1" a second time to the bits that are Set in the Uncorrectable Error and Correctable Error register clears the Status register (as in normal operation).

This feature allows the following Errors (Fatal, Non-Fatal, and Correctable) to be generated by software:

Table 1. Uncorrectable Error Status Register (FB8h in all Ports)

Bit	Register Description
4	Data Link Protocol Error Status
5	Surprise Down Error Status
12	Poisoned TLP Status
15	Completer Abort Status
16	Unexpected Completion Status
17	Receiver Overflow Status
18	Malformed TLP Status
19	ECRC Error Status
20	Unsupported Request Error Status
21	ACS Violation Status

Table 2. Correctable Error Status Register (FC4h in all Ports)

Bit	Register Description
0	Receiver Error Status
6	Bad TLP Status
7	Bad DLLP Status
8	REPLAY_NUM Rollover Status
12	Replay Timer Timeout Status
13	Advisory Non-Fatal Error Status

3. Software Generation of Hot Plug Interrupts (INTx, MSI or PEX_INTA#)

When the Software Force Error Enable bit (Ports 0 register 1C8h[2]) is Set in the switch, the Slot Status register bits that can generate associated interrupts (80h[24, 20:16]) bits are changed from RWIC to Read/Write (on alternate cycles). Then when software writes "1" to one or more of the implemented bits that are not already Set in the Slot Status register, the switch port will generate the corresponding Message (provided the corresponding Mask bit is not Set, and Messages are enabled in the Device Control register (70h)). Writing the value "1" a second time to the bits that are Set in the Uncorrectable Error and Correctable Error register clears the Status register (as in normal operation).

This feature allows the following Hot Plug events to be generated by software:

Table 3. Slot Status Register (80h in Transparent Ports)

Bit	Register Description
16	Attention Button Pressed
17	Power Fault Detected
18	MRL Sensor Changed
19	Presence Detect Changed
20	Command Completed
24	Data Link Layer State Changed

Additionally this feature allows Hot Plug interrupts to be generated by the switch upstream port, if needed for a particular application.

4. ECC RAM Error Injection

4.1 ECC RAMs that are Error Injectable

- Packet Link List RAM
- Payload RAM
- Header RAM
- Retry Buffer

All RAMs are instantiated separately in two half Stations; one half Station has Even Ports, i.e. Ports 0, 2, 4, and so on. Another half Station has Odd Ports, i.e. Ports 1, 3, 5, and so on.

4.2 ECC RAM Registers

Table 1 below lists the registers that are associated with the ECC RAMs that can be injected with 1-bit errors or 2-bit errors. These registers include Status bits, Mask bits (for enabling and disabling of the specific interrupts), 8-bit error counters for 1-bit ECC errors, and Injection bits to cause 1-bit and 2-bit ECC errors. 1-bit and 2-bit errors can be selectively injected into either the actual packet data, or into the ECC code itself.

The ECC RAM Error Injection bits, the Status and Mask bits, and the 8-bit Counters for 1-bit ECC errors exist in Port 0, except, if Port 0 is the Legacy NT port (STRAP_NT_P2P_EN# is high), then the registers exist in the NT-Virtual port (instead of Port 0).

Table 4. ECC RAM Register bits

Description	Counter	Status	Mask	Inject	Inject Data(1) / Code(0)
Even Port Packet Link List RAM 1-bit soft error	F28h[7:0]	1C0h[0]	1C4h[0]	660h[7]	660h[9]
Odd Port Packet Link List RAM 1-bit soft error	F28h[15:8]	1C0h[1]	1C4h[1]	660h[10]	660h[12]
Even Port Packet Link List RAM 2-bit soft error	-	1C0h[2]	1C4h[2]	660h[8]	660h[9]
Odd Port Packet Link List RAM 2-bit soft error	-	1C0h[3]	1C4h[3]	660h[11]	660h[12]
Even Port Payload RAM0 Instance0 1-bit soft error counter overflow	F18h[7:0]	1C0h[8]	1C4h[8]	660h[4]	660h[6]
Odd Port Payload RAM0 Instance 0 1-bit soft error counter overflow	F18h[23:16]	1C0h[9]	1C4h[9]	660h[19]	660h[21]
Even Port Payload RAM0 Instance1 1-bit soft error counter overflow	F18h[7:0]	1C0h[11]	1C4h[11]	660h[4]	660h[6]
Odd Port Payload RAM0 Instance 1 1-bit soft error counter overflow	F18h[23:16]	1C0h[12]	1C4h[12]	660h[19]	660h[21]
Even Port Payload RAM1 Instance 0 1-bit soft error counter overflow	F18h[15:8]	1C0h[20]	1C4h[20]	660h[4]	660h[6]
Odd Port Payload RAM1 Instance 0 1-bit soft error counter overflow	F18h[31:24]	1C0h[21]	1C4h[21]	660h[19]	660h[21]
Even Port Payload RAM1 Instance 1 1-bit soft error counter overflow	F18h[15:8]	1C0h[23]	1C4h[23]	660h[4]	660h[6]
Odd Port Payload RAM1 Instance 1 1-bit soft error counter overflow	F18h[31:24]	1C0h[24]	1C4h[24]	660h[19]	660h[21]
Even Port Payload RAM0 Instance 0 2-bit soft error	-	1C0h[14]	1C4h[14]	660h[5]	660h[6]
Odd Port Payload RAM0 Instance 0 2-bit soft error	-	1C0h[15]	1C4h[15]	660h[20]	660h[21]
Even Port Payload RAM0 Instance 1 2-bit soft error	-	1C0h[17]	1C4h[17]	660h[5]	660h[6]
Odd Port Payload RAM0 Instance 1 2-bit soft error	-	1C0h[18]	1C4h[18]	660h[20]	660h[21]
Even Port Payload RAM1 Instance 0 2-bit soft error	-	1C0h[26]	1C4h[26]	660h[5]	660h[6]
Odd Port Payload RAM1 Instance 0 2-bit soft error	-	1C0h[27]	1C4h[27]	660h[20]	660h[21]
Even Port Payload RAM1 Instance 1 2-bit soft error	-	1C0h[29]	1C4h[29]	660h[5]	660h[6]
Odd Port Payload RAM1 Instance 1 2-bit soft error	-	1C0h[30]	1C4h[30]	660h[20]	660h[21]
Retry Buffer 1-bit ECC error	-	1CCh[12]	1D0h[12]	F30h[8]	F30h[10]
Retry Buffer 2-bit ECC error	-	1CCh[13]	1D0h[13]	F30h[9]	F30h[10]
Even Port Header RAM 2-bit ECC error	-	1CCh[20]	1D0h[20]	660[14]	660[15]
Odd Port Header RAM 2-bit ECC error	-	1CCh[21]	1D0h[21]	660[17]	660[18]
Even Port Header RAM 1-bit error counter overflow	F24h[7:0]	1CCh[24]	1D0h[24]	660[13]	660[15]
Odd Port Header RAM 1-bit error counter overflow	F24h[15:8]	1CCh[25]	1D0h[25]	660[16]	660[18]

4.3 Hardware and Software Setup for Error Injection Demonstration

Error injection can be demonstrated using two PLX RDKs stacked together into one PCIe slot of a PC.

- The first RDK (such as an 8619 RDK) is the device to be tested and is plugged into the motherboard PCIe slot.
- The second RDK (such as an 8647 RDK) is used as a Packet Generator to generate traffic out of the switch upstream port, and is plugged into a downstream port slot on the first RDK (Port 5, for example). Error injection requires that traffic be sent through the switch after the injection bit is Set.

A second PC or laptop is used to read the registers of the first RDK, using I2C access (since in-band access to registers may not be possible following injection of a 2-bit error, which is fatal to the switch).

The PEX Device Editor (PDE software) included with the PLX SDK is used to access registers and operate the Packet Generator.

4.4 Error Injection Procedure Overview

- Packets that are generated from the downstream RDK (such as the 8647 RDK) are targeted to the PLX Buffer, which is created by the PLX PCI/PCIe Service Driver and resides in host memory.
- Traffic from the Packet Generator enters the downstream port of the device being tested, and the packet data is temporarily stored in the RAM for that Station.
- Payload RAM consists of two logical blocks of RAM, designated RAM0 and RAM1. Each block consists of two instances, designated Instance 0 and Instance 1. These RAMs are adjacent and are combined to form double-wide rows of RAM, with RAM0 containing lower addresses of the packet data and RAM1 containing higher relative addresses. The first data of each Payload is always stored in RAM0; if the payload length exceeds the width of RAM0, subsequent data of the same packet is stored into RAM1. Therefore to test Payload RAM1, Payload Length must be at least 64 bytes.
- Fatal (2-bit) ECC errors to Packet Link List RAM will also cause ECC error to Payload RAM, since 2-bit errors corrupt internal RAM pointers. When testing Packet Link List RAM, Payload Length should be at least 128 bytes.
- The RAM ECC is checked when the egress Port (Port 0 of the 8619 RDK, in this example) reads the RAM before the packet is to be forwarded (toward host memory in this example).
- When an Error Injection bit is Set (in the Port containing the ingress port), one ECC error is created; creation of an additional ECC error requires that the Error Injection bit be toggled (Cleared and then Set again).
- ECC errors can be injected into either the actual data or to the ECC code itself.
- 1-bit ECC errors increment the corresponding ECC Error Counter registers that exist in Port 0 (or the NT-Virtual port when Port 0 is a Legacy NT port). An interrupt will be generated when a counter overflows (provided interrupts are enabled). There is no counter for 1-bit errors to the Replay Buffer (an interrupt can be generated for each Replay Buffer ECC error).
- ECC error events that can generate interrupts (i.e, 1-bit ECC 8-bit error counter overflow, 2-bit ECC errors, and Replay Buffer 1-bit and 2-bit ECC errors), are logged in registers 1C0h and 1CCh in Port 0 (or the NT-Virtual port when Port 0 is a Legacy NT port).

    Note: MSI interrupts can be generated for prior events; if an error is flagged while MSI is not enabled, by default, subsequent enabling of MSI interrupts will trigger interrupt generation for the prior error event(s). In contrast, INTx and PEX_INTA# mechanisms will not generate interrupts for events that occurred prior to the enabling of interrupts. MSI, INTx and PEX_INTA# are mutually exclusive on a per-port basis, and the switch upstream port will not convert interrupts received from a downstream port to a different interrupt mechanism for egress out the switch upstream port.

- Following a fatal (2-bit) ECC error, the switch will require a reset (either a Hot Reset or a Fundamental Reset) to recover.

4.5 Error Injection Procedure

The following example demonstrates 1-bit ECC error injection to Payload RAM. This error will increment the counter but will only generate an interrupt when the counter overflows and interrupts are enabled.

Example: PEX 8647 RDK (used for packet generation from its upstream Port 0) is linked to the top slot on the PEX 8619 RDK (Port 1, belongs to Odd Port).

1. Payload RAM 1-bit ECC Error Counter in Port 0 register 0xF18[23:16] = 0
2. Set register 0x660[21] = 1. Setting this bit selects error injection into the data field (rather than the ECC code).
3. Set register 0x660[19] = 1 (Setting this bit enables error injection into the next packet that arrives from any ingress Port which is Odd Port)
4. Use the Packet Generator tool in the PDE utility to cause the PEX 8647 to send a Posted Write packet in the upstream direction to the PLX Buffer in system memory. Payload Length should be 64 bytes or 128 bytes (refer to section 4.4 above).
5. Read the Payload RAM 1-bit ECC Error Counter in Port 0 register 0xF18[23:16]. The count should increment by 1.
6. To inject ECC error again, clear the injection bit (register 0x660[19]) and repeat the procedure above beginning at step 3.

If 2-bit ECC errors are to be injected, use I2C to access the registers in steps 4 and 5, then reset the switch after each 2-bit error.