# Intel® Communications Chipset 89xx Series

## Specification Update

*October 2012*

**Document Number: 328000-001US**

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/#/en_US_01

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Go to: http://www.intel.com/products/processor%5Fnumber/

I2C is a two-wire communications bus/protocol developed by Philips. SMBus is a subset of the I2C bus/protocol and was developed by Intel. Implementations of the I2C bus/protocol may require licenses from various entities, including Philips Electronics N.V. and North American Philips Corporation.

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Copyright © 2012, Intel Corporation. All rights reserved.

# Contents

Revision History ..... 4
**Preface** ..... 5
Affected Documents/Related Documents ..... 5
Nomenclature ..... 5
**Summary Tables of Changes** ..... 6
Codes Used in Summary Tables ..... 6
Stepping ..... 6
Page ..... 6
Status ..... 6
Row ..... 6
**Identification Information** ..... 11
Component Marking Information ..... 11
**Device & Revision Identification** ..... 12
**Errata** ..... 13
**Specification Changes** ..... 36
**Specification Clarifications** ..... 37
**Documentation Changes** ..... 39

# Revision History

| Date | Revision | Description |
|---|---|---|
| October 2012 | 001 | • Initial Release |

# Preface

This document is an update to the specifications contained in the Affected Documents/ Related Documents table below. This document is a compilation of device and documentation errata, specification clarifications and changes. It is intended for hardware system manufacturers and software developers of applications, operating systems, or tools.

Information types defined in Nomenclature are consolidated into the specification update and are no longer published in other documents.

This document may also contain information that was not previously published.

## Affected Documents/Related Documents

| Document Title | Document Number/ Location |
|---|---|
| *Intel® Communications Chipset 89xx Series - Datasheet* | 327879-00x |

## Nomenclature

**Errata** are design defects or errors. Errata may cause the behavior of the PCH to deviate from published specifications. Hardware and software designed to be used with any given stepping must assume that all errata documented for that stepping are present in all devices.

**Specification Changes** are modifications to the current published specifications. These changes will be incorporated in any new release of the specification.

**Specification Clarifications** describe a specification in greater detail or further highlight a specification's impact to a complex design situation. These clarifications will be incorporated in any new release of the specification.

**Documentation Changes** include typos, errors, or omissions from the current published specifications. These will be incorporated in any new release of the specification.

# Summary Tables of Changes

The following tables indicate the errata, specification changes, specification clarifications, or documentation changes which apply to the Intel® Communications Chipset 89xx Series product. Intel may fix some of the errata in a future stepping of the component, and account for the other outstanding issues through documentation or specification changes as noted. These tables uses the following notations:

## Codes Used in Summary Tables

### Stepping

| | |
|---|---|
| X: | Errata exists in the stepping indicated. Specification Change or Clarification that applies to this stepping. |
| (No mark) | |
| or (Blank box): | This erratum is Fixed in listed stepping or specification change does not apply to listed stepping. |

### Page

| | |
|---|---|
| (Page): | Page location of item in this document. |

### Status

| | |
|---|---|
| Doc: | Document change or update will be implemented. |
| Future Fix: | This erratum may be Fixed in a future stepping of the product. |
| Fixed. | This erratum has been previously Fixed. |
| No Fix: | There are no plans to fix this erratum. |

### Row

Change bar to left of table row indicates this erratum is either new or modified from the previous version of the document.

## Table 1. Errata (Sheet 1 of 3)

| Erratum Number | Stepping | Status | ERRATA |
|---|---|---|---|
| | C1 | | |
| 1 | X | No Fix | DMI Port: DMI Degradation Mode not Functional if Lane Reversal is implemented |
| 2 | X | No Fix | PCIe Root Port (RP): PCIe RP Degradation Mode not Functional if Lane Reversal is implemented |
| 3 | X | No Fix | EndPoint (EP): PCIe EP does not check the PCIe TLP Digest (TD) bit |
| 4 | X | No Fix | GbE: GbE SMGII Protocol Layer Delays affects System Collision Detection |
| 5 | X | No Fix | GbE: GbE MAC may drop the first Receive (RX) packet after a Link Speed Change |
| 6 | X | No Fix | PCIe Root Port (RP): PCIe RP Degradation is not Functional. |
| 7 | X | No Fix | SATA: SATA Low Power Device Detection |
| 8 | X | No Fix | USB: USB Port Stall with Bulk and Control Traffic |
| 9 | X | No Fix | SATA: SATA SYNC Escape Issue |
| 10 | X | No Fix | USB: USB End of Frame When Retrying Packets Issue |
| 11 | X | No Fix | USB: USB Classic Device Removal Issue |
| 12 | X | No Fix | USB: USB RMH Descriptor May Report Incorrect Number of USB Ports |
| 13 | X | No Fix | PCH: High Precision Event Timer (HPET) Writing Timing Issue |
| 14 | X | No Fix | USB: USB Full-Speed Port Staggering |
| 15 | X | No Fix | USB: USB Devices May Slow or Hang |
| 16 | X | No Fix | USB: USB Low-Speed Control Transactions |
| 17 | X | No Fix | SATA: SATA Controller May Not Detect Unsolicited SATA COMINITs |
| 18 | X | No Fix | SATA: SATA Hot Unplug May Not be Detected |
| 19 | X | No Fix | USB: USB Missing ACK |
| 20 | X | No Fix | SATA: SATA 6 Gb/s Device Detection |
| 21 | X | No Fix | PCIe Root Port (RP): PCIe RP Link Disable Bit |
| 22 | X | No Fix | USB: USB Isochronous In Transfer Error Issue |
| 23 | X | No Fix | USB: USB Full-Speed/Low-Speed Device Removal Issue |
| 24 | X | No Fix | USB: USB Babble Detected with Software Overscheduling |
| 25 | X | No Fix | USB: USB Low-Speed/Full-Speed EOP Issue |
| 26 | X | No Fix | USB: USB PLL Control FSM not Getting Reset on Global Reset |
| 27 | X | No Fix | USB: USB Asynchronous Retries Prioritized Over Periodic Transfers |
| 28 | X | No Fix | USB: Incorrect Data for Low-Speed or Full-Speed USB Periodic IN Transaction |
| 29 | X | No Fix | USB: USB RMH Delayed Periodic Traffic Timeout Issue |
| 30 | X | No Fix | GbE: GbE MNG Reset Clears Resource Grant With No Feedback |
| 31 | X | No Fix | GbE MDIO: Com_MDIO and Destination Bits of MDICNFG Register Are Not Loaded Consistently from EEPROM |
| 32 | X | No Fix | GbE MNG: I2C Data Out Hold Time Violation |
| 33 | X | No Fix | GbE EEPROM: EE_CS_N Control Signal Hold Time Violation |
| 34 | X | No Fix | GbE SGMII: Counters Incorrectly Increment on Collision |
| 35 | X | No Fix | GbE TSYNC: Auxiliary Timestamp from SDP is Unreliable |
| 36 | X | No Fix | EndPoint (EP): Spurious MSI/MSIX Interrupt Generated |
| 37 | X | No Fix | EndPoint (EP): PCIe EP May not Detect Unexpected Completion Packets |

## Table 1. Errata (Sheet 2 of 3)

| Erratum Number | Stepping | Status | ERRATA |
|---|---|---|---|
| | C1 | | |
| 38 | X | No Fix | GbE: GbE Near End Analog Loopback Not Supported |
| 39 | X | No Fix | PCIe Root Port (RP): PCIe RP May not Automatically Switch into Compliance Mode |
| 40 | X | No Fix | EndPoint (EP): PCIe EP Incorrect Completion ID |
| 41 | X | No Fix | EndPoint (EP): PCIe EP Link Status Register 2 (LNKSTS2) Not Implemented for Functions[1-4] |
| 42 | X | No Fix | EndPoint (EP): PCIe Malformed Packet Checking is not implemented |
| 43 | X | No Fix | EndPoint (EP): PCIe EP Virtual Functions Wrongly Reporting Advisory Errors as Correctable Errors |
| 44 | X | No Fix | EndPoint (EP): PCIe EP Sends Two Error Messages to the Host for IO_RD to Unsupported Address Space |
| 45 | X | No Fix | EndPoint (EP): PCIe EP May Generate Spurious Error Message |
| 46 | X | No Fix | EndPoint (EP): PCIe EP Incorrect Handling of Multiple Errors |
| 47 | X | No Fix | EndPoint (EP): PCIe EP Incorrect Handling of Poisoned Memory Writes. |
| 48 | X | No Fix | EndPoint (EP): PCIe EP Does Not Detect Poisoned Data for IO_WR Transactions |
| 49 | X | No Fix | EndPoint (EP): PCIe EP Does Not Reliably Update the Header Log and First Error Pointer |
| 50 | X | No Fix | EndPoint (EP): PCIe EP Does Not Report the Function Number Associated with a Poisoned TLP Error |
| 51 | X | No Fix | EndPoint (EP): PCIe EP Does Not Send Error Message for MMIO Read transaction to Unsupported Memory Space |
| 52 | X | No Fix | EndPoint (EP): Active State Power Management (ASPM) Not Supported |
| 53 | X | No Fix | EndPoint(EP): Incorrect Logging & Reporting of Data Link Protocol Errors (DLPEs) |
| 54 | X | No Fix | EndPoint (EP): PCIe EP Transition from D3Hot to L1 |
| 55 | X | No Fix | EndPoint (EP): Some PCIe EP Configuration Registers have Wrong Attribute Assignment |
| 56 | X | No Fix | EndPoint (EP): PCIe EP Transition from D3hot to D0uninitialized |
| 57 | X | No Fix | EndPoint (EP): PCIe EP Link Control Register 2 (PLCNTLR2) Sticky Bits Not Retaining value after Hot Reset |
| 58 | X | No Fix | USB: PLL Configuration Settings for USB-IO Interface |
| 59 | X | No Fix | EndPoint (EP): PCIe Transaction Pending Bit Initialization |
| 60 | X | No Fix | EndPoint (EP): Incorrect IO Transaction Response in D3hot State |
| 61 | X | No Fix | PCH: IEEE Std. 1149.6 EXTEST_PULSE and EXTEST_TRAIN Instructions Not Supported by Non-EP_JTAG Port. |
| 62 | X | No Fix | PCH: Auxiliary (AUX) and Auxiliary2 (AUX2) Thermal Trip Interrupts not Functional. |
| 63 | X | No Fix | EndPoint (EP): PCIe EP Does Not Report Timeout Errors |
| 64 | X | No Fix | EndPoint (EP): PCIe EP Does Not Register Correctable Errors |
| 65 | X | No Fix | EndPoint (EP): PCIe EP Incorrect Handling of Read Completions with Unsupported Request (UR) Status |
| 66 | X | No Fix | EndPoint (EP): PCIe EP does not Enable the Correct number of Virtual Functions (VFs) |
| 67 | X | No Fix | GbE: SGMII Interface Receiver does not meet Input Differential Hysteresis (Vhyst) Specification |
| 68 | X | No Fix | SATA: Incorrect Number of Supported Ports Reported |
| 69 | X | No Fix | Endpoint (EP): PCIe EP Does not Indicate Poisoned Data from Reads of Corrupted Memory |

## Table 1. Errata (Sheet 3 of 3)

| Erratum Number | Stepping | Status | ERRATA |
|---|---|---|---|
| | C1 | | |
| 70 | X | No Fix | Endpoint (EP): PCIe EP Link May Not Train |
| 71 | X | No Fix | GbE: SGMII Interface Transmit Rise ($t_{rise}$) and Fall ($t_{fall}$) AC Timing Specification Violation |
| 72 | X | No Fix | QuickAssist Technology (QAT): Decompression Adler32 Checksum May be Calculated Incorrectly |
| 73 | X | No Fix | Endpoint (EP): PCIe EP Link Disable requires Hot Reset |
| 74 | X | No Fix | GbE: GbE MACs PCI Config Space Does not contain Valid VID/DID Default Values |
| 75 | X | No Fix | GbE: GbE Interface Transmit Voltage Level Specification Violation |
| 76 | X | No Fix | USB: USB Full/Low Speed Port Reset or Clear Transaction Translation (TT) Buffer Request |
| 77 | X | No Fix | USB: USB RMH Think Time Issue |
| 78 | X | No Fix | USB: USB RMH False Disconnect Issue |

## Table 2. Specification Changes

| Number | Specification Change |
|---|---|
| | None to report at this time. |

## Table 3. Specification Clarifications

| Number | Specification Clarification |
|---|---|
| 1 | GbE: Use of Wake on LAN Together with Manageability |
| 2 | GbE SMBus: Illegal STOP Condition |
| 3 | GbE SERDES: AN_TIMEOUT Only Works When Link Partner Idle |

## Table 4. Documentation Changes

| Number | Documentation Change |
|---|---|
| | None to report at this time. |

# Identification Information

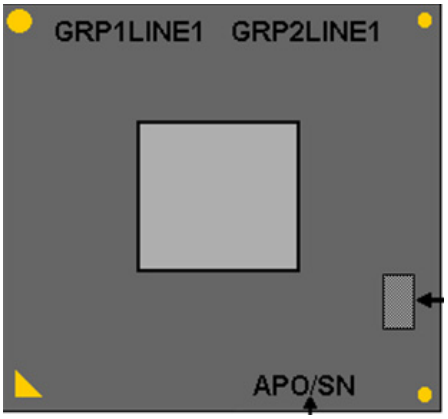
## Component Marking Information

The Intel® Communications Chipset 89xx Series PCH components are identified in the component markings in Table 5.

### Table 5. Component Identification

| PCH Stepping | MM# | S-Spec (GRP1LINE1) | Product | Notes |
|---|---|---|---|---|
| C1 | 923820 | SLJW2 | DH8900CC - SKU1 | SKU1: Intel® Communications Chipset 8900 |
| | 923819 | SLJVZ | DH8903CC - SKU2 | SKU2: Intel® Communications Chipset 8903 |
| | 923818 | SLJVY | DH8910CC - SKU3 | SKU3: Intel® Communications Chipset 8910 |
| | 923817 | SLJVX | DH8920CC - SKU4 | SKU4: Intel® Communications Chipset 8920 |

### Figure 1. Top Markings



# Device & Revision Identification

The Revision ID (RID) is traditionally an 8-bit register located at the offset 08h in the PCI header of every PCI device and function. The assigned value is based on the product stepping.

**Table 6. Intel® Communications Chipset 89xx Series Device and Revision ID Table**

| Device Function | PCH PCIe Devices | Device ID (DID) | Revision ID (RID) | Comments |
|---|---|---|---|---|
| B0:D28:fn | | 0x233E | | Used by the PCIe Root Port function that has subtractive decode compatibility set to '1'. |
| B0:D31:F0 | LPC | 0x2310 | 0x08 | LPC Controller |
| B0:D31:F2 | SATA | 0x2323 | 0x08 | SATA Controller #1 |
| B0:D31:F3 | SMBus | 0x2330 | 0x08 | SMBus Host Controller |
| B0:D31:F5 | SATA | 0x2326 | 0x08 | SATA Controller #2 |
| B0:D31:F6 | Thermal | 0x2332 | 0x08 | Thermal Subsystem |
| B0:D31:F7 | WDT | 0x2360 | 0x08 | WDT for Core Reset |
| B0:D29:F0 | USB | 0x2334 | 0x08 | USB Gen 2 |
| B0:D22:F0 | Intel® MEI #1 | 0x2364 | 0x08 | Intel® Management Engine Interface #1 |
| B0:D22:F1 | Intel® MEI #2 | 0x2365 | 0x08 | Intel® Management Engine Interface #2 |
| B0:D28:F0 | PCIe Root Port 1 | 0x2342 | 0x08 | The PCIe Root Port with a subtractive decode will have a different Device ID (See Row 1 of this table). |
| B0:D28:F1 | PCIe Root Port 2 | 0x2344 | 0x08 | |
| B0:D28:F2 | PCIe Root Port 3 | 0x2346 | 0x08 | |
| B0:D28:F3 | PCIe Root Port 4 | 0x2348 | 0x08 | |
| BM:D0:F0 | PCIe EndPoint | 0x0434 | 0x21 | PCIe Endpoint and QuickAssist Technology (QAT) |
| BM:D0:F1 | GbE MAC 0 | 0x0436 | 0x21 | The Device ID for the GbE Controllers can be overwritten by the EEPROM. |
| BM:D0:F2 | GbE MAC 1 | 0x0436 | 0x21 | |
| BM:D0:F3 | GbE MAC 2 | 0x0436 | 0x21 | |
| BM:D0:F4 | GbE MAC 3 | 0x0436 | 0x21 | |

# Errata

## 1. DMI Port: DMI Degradation Mode not Functional if Lane Reversal is implemented

Problem: DMI Degradation mode will not work when DMI lanes are routed in reverse order.

Implication: If the DMI Port is routed with the lanes reversed, it will work in full-mode only. The interface will not work if a link problem occurs and causes the interface to degrade.

Workaround: Do not route DMI lanes in reverse order.

Status: No Fix.

## 2. PCIe Root Port (RP): PCIe RP Degradation Mode not Functional if Lane Reversal is implemented

Problem: PCIe RP Degradation mode will not work when the PCIe lanes are routed in reverse order.

Implication: If the PCIe RP is routed with the lanes reversed, it will work in full-mode only. The interface will not work if an error occurs and causes the interface to degrade.

Workaround: Do not route PCIe RP lanes in reverse order.

Status: No Fix

## 3. EndPoint (EP): PCIe EP does not check the PCIe TLP Digest (TD) bit

Problem: The EP does not check the TD bit (bit 7 of byte 2) in the PCIe Transaction Layer Packet (TLP) header.

Implication: If the EP receives a TLP with the TD bit set, the following will happen:

- If the TLP contains the 4-byte digest field, the EP will not complete the request and will incorrectly respond with a Malformed Packet completion.
- If the TLP does not contain the 4-byte digest field, the EP will complete the request instead of responding with a Malformed Packet completion.

Workaround: Ensure that the TD bit in the TLP header is not set.

Status: No Fix.

## 4. GbE: GbE SMGII Protocol Layer Delays affects System Collision Detection

Problem: GbE SGMII protocol delays inhibits the proper detection of system collisions.

Implication: Collision detection errors may cause transmission data corruption.

Workaround: Reduce standard network transmission wire length by 10% to offset delays introduced by the SMGII Layer.

Status: No Fix.

## 5. GbE: GbE MAC may drop the first Receive (RX) packet after a Link Speed Change

Problem: The GbE MAC internal clock synchronization may not be complete before the SGMII link is established following a link speed change. This may result in the first Receive (RX) packet immediately following the link speed change to be truncated in the MAC layer and subsequently dropped due to bad Frame Check Sequence (FCS).

Implication: The Ethernet protocol FCS protects any bad/incomplete packet that may have appeared in MAC as an effect of this issue. Dropped packet due to bad FCS may be recorded in statistics.

Workaround: None.

Status: No Fix.

## 6. PCIe Root Port (RP): PCIe RP Degradation is not Functional.

Problem: The PCIe RP cannot degrade from a x4 link to a x2 or x1 link.

Implication: If the RP is configured as PCIe x4 port and a x2 or x1 device is connected, the link will not train and the device will not be seen.

Workaround: Using the Flash Image Configuration Tool (FITC), configure Soft Strap 9 to bifurcate the root port as 4x1 or 1x4 as needed on the platform.

Status: No Fix.

## 7. SATA: SATA Low Power Device Detection

Problem: The SATA Low Power Device Detection (SLPD) may not recognize, or may falsely detect, a SATA hot-plug event during a Partial or Slumber Link Power Management (LPM) state.

Implication: On systems that enable LPM, when a SATA device attached to the PCH is configured as External or hot-plug capable, one of the following symptoms may occur:

- Symptom #1: A hot-plug or External SATA device removal which is not detected results in the OS and Intel® Matrix Storage Manager or Intel® Rapid Storage Technology console falsely reporting the device present, or incorrectly identifying an eSATA device.
- Symptom#2: A false hot-plug removal detection may occur resulting in OS boot hang or ODD media playback hang.

Workaround: A Software Driver workaround is implemented.

Status: No Fix.

## 8. USB: USB Port Stall with Bulk and Control Traffic

Problem: When a single USB bulk device is active on an EHCI controller, and the device has pending control and bulk traffic, the USB controller may not be able to resolve which traffic type is a priority and the association with the device may stall.

The processor must be in C0 for an extended period of time, such as when Cx states are disabled, or if system traffic prevents the system from leaving C0.

Implication: The USB device may appear unresponsive. If Cx states are enabled, the device may recover a short time later.

***Note:*** Intel has only observed this failure on a limited number of devices. Failure only occurs if software associated with a USB device programs the Nak Count Reload bits defined in the EHCI Specification for USB Rev 1.0 to 0.

Workaround: BIOS implementation follows Intel® Communications Chipset 89xx Series BIOS Specification and program D29:F0:88h[7] = "1b".

Status: No Fix.

## 9. SATA: SATA SYNC Escape Issue

Problem: When SYNC Escape by a SATA device occurs on a D2H FIS, the Chipset 89xx Series does not set the PxIS.IFS bit to '1.' This deviates from section 6.1.9 of the Rev 1.3 Serial ATA Advanced Host Controller Interface (AHCI).

Implication: There is no known observable impact. Instead of detecting the IFS bit, software will detect a timeout error caused by the SYNC escape and then respond.

Workaround: None.

Status: No Fix.

## 10. USB: USB End of Frame When Retrying Packets Issue

Problem: If the USB controller encounters a full-speed or low-speed USB transaction with errors, it may retry the transaction without considering if the transaction can finish before the end of the current frame.

Implication: The implication depends on the particular USB device. The USB controller will attempt to recover per error handling specified in Section 4.5.2 of the USB Specification 2.0. The device may hang and require cycle to resume normal functionality.

***Note:*** Intel has only observed this behavior on a limited number of USB devices. The implication only occurs if a USB device does not correctly respond to error handling as specified in Section 4.5.2 of the USB Specification 2.0.

Workaround: None.

Status: No Fix.

## 11. USB: USB Classic Device Removal Issue

Problem: If two or more USB full-speed / low-speed devices are connected to the same USB controller, and if the devices are not suspended, then if one device is removed, one or more of the devices remaining in the system may be affected by the disconnect.

Implication: The implication is device-dependent. A device may experience a delayed transaction, stall, and be recovered via software, or it may stall and require a reset such as a hot-plug to resume normal functionality.

Workaround: None.

Status: No Fix.

## 12. USB: USB RMH Descriptor May Report Incorrect Number of USB Ports

Problem: The PCH supports 6 ports on RMH #1 and may incorrectly report 8 USB ports in the bNbrPorts field of the RMH hub descriptor.

Implication: If AC power is removed while the system is in hibernate, when the system resumes , new USB devices may not be detected, and all devices on RMH #1 may not function.

*Note: AC power removal while the system is in hibernate is not a normal usage model, or common occurrence.*

Workaround: None.

Status: No Fix.

## 13. PCH: High Precision Event Timer (HPET) Writing Timing Issue

Problem: A read transaction that immediately follows a write transaction to the HPET register space may return an incorrect value.

Implication: Implication depends on the usage model as noted below:

- For the HPET TIMn_COMP Timer 0 Comparator Value Register and HPET MAIN_CNT—Main Counter Value Register, the issue could result in the software receiving stale data. This may result in undetermined system behavior.

  ***Note:*** Timers [1:7] are not affected by this issue.
- For TIMERn_VAL_SET_CNF bit 6 in the TIMn_CONF—Timer n Configuration, there is no known usage model for reading this bit and there are no known functional implications.
- A write to the High Precision Timer Configuration (HPTC) register followed by a read to HPET register space, may return all 0xFFFF_FFFFh.

Workaround: Software workaround has been identified as described below:

- A write to the HPET TIMn_COMP Timer 0 Comparator Value Register should be followed by two reads that are discarded, and a third read where the data can be used.
- A write to the HPET MAIN_CNT - Main Counter Register should be followed by one read that is discarded, and a second read where the data can be used.

Workaround: TIMERn_VAL_SET_CNF bit 6 in the TIMn_CONF - Timer n - There is no known usage model to read this bit, but a write to the bit should be followed by one read that is discarded and a second read where the data can be used.

Status: No Fix.

## 14. USB: USB Full-Speed Port Staggering

Problem: When USB full-speed/low-speed port staggering is enabled, the USB controller may not wait for the bus to return to an idle state after an End of Packet (EOP), and may incorrectly acknowledge bus noise as a data packet.

Implication: Some full-speed/low-speed devices may fail to enumerate and function.

> ***Note:*** This issue has been seen with a minimum number of devices on some motherboard ports with certain cable and trace lengths.

Workaround: BIOS should disable USB FS/LS Port Staggering by clearing RCBA+3564h[12].

Status: No Fix.

## 15. USB: USB Devices May Slow or Hang

Problem: When the processor is in C0, and a single bulk high-speed USB device is active, the port associated with the active device may hang.

Implication: The implication is device driver-dependent. Intel has observed some USB devices may have decreased performance, or the device may hang.

Workaround: BIOS implementation follows Intel® Communications Chipset 89xx Series BIOS Specification (section 16.9) and programs D29:F0:88h[7]= “1b“.

Status: No Fix.

## 16. USB: USB Low-Speed Control Transactions

Problem: If the USB control buffers in the PCH Rate Matching Hub(s) are saturated with pending transactions, the buffers may not be serviced in round robin order.

Implication: Some low-speed endpoints may not receive their pending control transactions.

> ***Note:*** This issue has only been observed in a synthetic test environment. The implication will be device, driver, and operating system specific.

Workaround: None.

Status: No Fix.

## 17. SATA: SATA Controller May Not Detect Unsolicited SATA COMINITs

Problem: SATA controller may not detect an unsolicited COMINIT from a SATA device.

Implication: The SATA device may not be properly detected and configured, resulting in the device not functioning as expected.

Workaround: BIOS implementation follows Intel® Communications Chipset 89xx Series BIOS Specification and programs D31:F2:Offset 98h[20:19] = “11b”.

Status: No Fix.

## 18. SATA: SATA Hot Unplug May Not be Detected

Problem: SATA controller may not detect the unplug of a SATA 3.0 Gb/s device on a hot-plug enabled SATA port.

Implication: The unplugged SATA device may temporarily appear to be available.

Workaround: BIOS should program D31:F2:Offset 98h[6:5] = “00b” .

Status: No Fix.

## 19. USB: USB Missing ACK

Problem: Following system power cycling or S3-S5 resume, if both high-speed and low-speed/full-speed devices are attached to the same controller, the host controller may not respond to a high-speed device ACK during a Get Descriptor request from the host software to a USB high-speed port.

Implication: USB high-speed devices may not be detected after a power cycling or S3-S5 resume.

- Intel has only observed this failure on a limited number of platforms. On a failing platform, the issue occurs infrequently.
- Full-speed and low-speed USB devices are not impacted by this issue.

Workaround: None.

Status: No Fix.

## 20. SATA: SATA 6 Gb/s Device Detection

Problem: The SATA controller may not be able to complete SATA Out Of Band (OOB) signaling with SATA 6Gb/s devices and down-shift to SATA 3 Gb/s speed.

Implication: SATA controller may not detect a SATA 6 Gb/s device upon power up or resume from S3, S4 or S5 State, resulting in indeterminate system behavior.

Workaround: None.

Status: No Fix.

## 21. PCIe Root Port (RP): PCIe RP Link Disable Bit

Problem: The PCIe RP may not exit the disable state when the Link Control Register "Link Disable" bit is set and PCIe Device Electrical Idle Exit is detected.

Implication: Port-specific software-directed Hot Plug or Power Management (PM) support using the "Link Disable" bit may cause PCIe RP to be stuck in the "Link Disable state" until a Host Reset with Power Cycling occurs.

Workaround: For PCIe RP port-specific software-directed Hot Plug or Power Management support, use the PCI Power Management Control register D3HOT bits instead of the Link Disable bit.

Status: No Fix.

## 22. USB: USB Isochronous In Transfer Error Issue

Problem: If a USB full-speed inbound isochronous transaction with a packet length 190 bytes or greater is started near the end of a micro-frame, the USB controller may see more than 189 bytes in the next micro-frame.

Implication: If the USB controller sees more than 189 bytes for a micro-frame, an error will be sent to software and the isochronous transfer will be lost. If a single data packet is lost, no perceptible impact for the end user is expected.

***Note:*** Intel has only observed the issue in a synthetic test environment where precise control of packet scheduling is available, and has not observed this failure in its compatibility validation testing.

- Isochronous traffic is periodic and cannot be retried, thus it is considered good practice for software to schedule isochronous transactions to start at the beginning of a micro-frame. Known software solutions follow this practice.

- To sensitize the system to the issue, additional traffic such as other isochronous transactions or retries of asynchronous transactions would be required to push the inbound isochronous transaction to the end of the micro-frame.

Workaround: None.

Status: No Fix.

## 23. USB: USB Full-Speed/Low-Speed Device Removal Issue

Problem: If two or more USB full-speed/low-speed devices are connected to the same USB controller, the devices are not suspended, and one device is removed, one or more of the devices remaining in the system may be affected by the disconnect.

Implication: The implication is device-dependent. A device may experience a delayed transaction, stall, and be recovered via software, or it may stall and require a reset such as a hot-plug to resume normal functionality.

Workaround: None.

Status: No Fix.

## 24. USB: USB Babble Detected with Software Overscheduling

Problem: If software violates USB periodic scheduling rules for Full-Speed isochronous traffic by overscheduling, the RMH may not handle the error condition properly and return a completion split with more data than the length expected.

Implication: If the RMH returns more data than expected, the endpoint will detect packet babble for that transaction, and the packet will be dropped. Since overscheduling occurred to create the error condition, the packet would be dropped regardless of RMH behavior. If a single isochronous data packet is lost, no perceptible impact to the end user is expected.

***Note:*** USB software overscheduling occurs when the amount of data scheduled for a micro-frame exceeds the maximum budget. This is an error condition that violates the USB periodic scheduling rule.

***Note:*** This failure has only been recreated synthetically with USB software intentionally overscheduling traffic to hit the error condition.

Workaround: None.

Status: No Fix.

## 25. USB: USB Low-Speed/Full-Speed EOP Issue

Problem: If the EOP of the last packet in a USB Isochronous split transaction (defined as a transaction > 189 bytes) is dropped or delayed 3 ms or longer, the following may occur:

- If there are no other pending low-speed or full-speed transactions, the RMH will not send SOF, or Keep-Alive. Devices connected to the RMH will interpret this condition as idle and will enter suspend.
- If there are other pending low-speed or full-speed transactions, the RMH will drop the isochronous transaction and resume normal operation.

Implication:

- If there are no other transactions pending, the RMH is unaware a device has entered suspend and may start sending a transaction without waking the device.

The implication is device-dependent, but a device may stall and require a reset to resume functionality.

- If there are other transactions present, only the initial isochronous transaction may be lost. The loss of a single isochronous transaction may not result in end user perceptible impact.

  ***Note:*** Intel has only observed this failure when using software that does not comply with the USB specification and violates the hardware isochronous scheduling threshold by terminating transactions that are already in progress.

Workaround: None.

Status: No Fix.

## 26. USB: USB PLL Control FSM not Getting Reset on Global Reset

Problem: The USB PLL may not lock if a Global Reset occurs early during a cold boot sequence.

Implication: The USB Port would not be functional and additional cold boot would be necessary to recover.

Workaround: None.

Status: No Fix.

## 27. USB: USB Asynchronous Retries Prioritized Over Periodic Transfers

Problem: The integrated USB RMH incorrectly prioritizes low-speed and full-speed asynchronous retries over dispatchable periodic transfers.

Implication: Periodic transfers may be delayed or aborted. If the asynchronous retry latency causes the periodic transfer to be aborted, the impact varies depending on the nature of periodic transfer:

- If a periodic interrupt transfer is aborted, the data may be recovered by the next instance of the interrupt or the data could be dropped.
- If a periodic isochronous transfer is aborted, the data will be dropped. A single dropped periodic transaction should not be noticeable by the end user.

  ***Note:*** This issue has only been seen in a synthetic environment. The USB spec does not consider the occasional loss of periodic traffic a violation.

Workaround: None.

Status: No Fix.

## 28. USB: Incorrect Data for Low-Speed or Full-Speed USB Periodic IN Transaction

Problem: The Periodic Frame list entry in DRAM for a USB low-speed or full-speed Periodic IN transaction may incorrectly get some of its data from a prior Periodic IN transaction which was initiated very late into the preceding micro-frame.

It is considered good practice for software to schedule Periodic Transactions at the start of a micro-frame. However Periodic transactions may occur late into a micro-frame due to the following cases:

- Asynchronous transaction starting near the end of the proceeding micro-frame gets asynchronously retried.

***Note:*** Transactions getting asynchronous retried would only occur for ill-behaved USB device or USB port with a signal integrity issue.

- Two Periodic transactions are scheduled by software to occur in the same micro-frame and the first one needs to push the second Periodic IN transaction to the end of the micro-frame boundary

Implication: The implication will be device, driver, or operating system specific.

***Note:*** This issue has only been observed in a synthetic test environment.

Workaround: None.

Status: No Fix.

## 29. USB: USB RMH Delayed Periodic Traffic Timeout Issue

Problem: If an interrupt transaction is pushed to the x+4 micro-frame boundary due to asynchronous retries, the RMH may not wait for the interrupt transaction to time out before starting the next transaction.

IF RMH Transaction Translator (TT) reaches a discard boundary, a timeout may be ignored.

Implication: If the next transaction is intended for the same device targeted by the interrupt, the successful completion of that transaction is device dependent and cannot be guaranteed. The implication may differ depending on the nature of the transaction: (this only impacts TT – low-speed / full-speed).

- If the transaction is asynchronous and the device does not respond, it will eventually be retried with no impact.
- If the transaction is periodic and the device does not respond, the transfer may be dropped. A single dropped periodic transaction should not be noticeable by the end user.

***Note:*** This issue has only been seen in a synthetic environment.

Workaround: None.

Status: No Fix.

## 30. GbE: GbE MNG Reset Clears Resource Grant With No Feedback

Problem: When accessing the EEPROM (via EEC register), grants may be lost due to deadlock or firmware reset. Software will not be notified of the lost grant. A driver in the middle of a bit bang may renew the request and receive the grant without knowing that it is actually starting a new transaction.

Implication:

1. EEC bit banging transactions may fail.
2. Long transactions may turn into different transactions than expected.

Workaround:

1. Software should not execute bit bang sequences longer than one word at a time.
2. When software reads the EEC, it should make sure that it still has the request and grant. If not, software should renew it and re-start the transaction. This does not cover all cases but reduces the possibility of a problem.

Status: No Fix.

## 31. GbE MDIO: Com_MDIO and Destination Bits of MDICNFG Register Are Not Loaded Consistently from EEPROM

Problem: The Com_MDIO (bit 30) and Destination (bit 31) bits of the MDICNFG register (0x0E04) are not loaded consistently from the EEPROM. In some cases, the hardware default value of 0b is used instead.

Implication: Software that relies on the EEPROM-loaded value might not be able to initialize an external PHY.

Workaround: Software should assume that the initial values of these bits are undefined and should program them before attempting to initialize an external PHY. The EEPROM bits can be used to determine the intended settings.

***Note:*** A GbE driver workaround has been identified and deployed for Intel drivers.

Status: No Fix.

## 32. GbE MNG: I2C Data Out Hold Time Violation

Problem: The GbE MNG I2C interface should provide a data out hold time of 50 ns on the SFPx_I2C_DATA pins. The actual hold time is about 16 ns.

Implication: I2C timing specification violation. There have been no reports of failures resulting from this timing. Note that the data input hold time required is zero, so the provided output hold time should be more than enough as long as the I2C CLK and DATA signals are reasonably matched on the board.

Workaround: None.

Status: No Fix.

## 33. GbE EEPROM: EE_CS_N Control Signal Hold Time Violation

Problem: The EEPROM datasheet indicates a hold time of 250 ns for EE_CS_N relative to the falling edge of EE_SK (tCSH). At the end of a READ or RDSR operation, EE_CS_N is actually negated about 32 ns after the falling edge of EE_SK.

Implication: Although this is a timing specification violation for many EEPROM devices, no malfunction has been reported.

Workaround: None.

Status: No Fix.

## 34. GbE SGMII: Counters Incorrectly Increment on Collision

Problem: In SGMII mode/half duplex, the statistics counters listed below incorrectly increment when a collision occurs:

| Name | Definition | Location |
|---|---|---|
| RLEC | Length error counter | 0X4040 |
| CRCERRS | CRC error counter | 0x4000 |
| RFC | receive frame counter | 0x40A8 |

Implication: Error counters may not be accurate.

Workaround: None.

Status: No Fix.

## 35. GbE TSYNC: Auxiliary Timestamp from SDP is Unreliable

Problem: The SDP inputs to the timestamp logic are not properly synchronized. As a result, both the Auxiliary Timestamp Register values and the Auxiliary Timestamp Taken bits in TSAUXC are sometimes loaded incorrectly.

Implication: The auxiliary timestamp feature should be considered unreliable.

Workaround: For applications that use the auxiliary timestamp feature to synchronize to an external clock, it might be acceptable to drop some of the samples. For such applications, software can filter out many of the incorrect timestamp values by comparing them to an approximate expected timestamp and discarding unreasonable values.

In addition, the following method can be used to filter out incorrect values:

- Connect the input signal to two SDP inputs for the same port.
- Using the TSSDP register, assign one of the SDP inputs to AUX0 and the other SDP input to AUX1.
- When reading the TSAUXC register to check for new samples, check that both AUTT0 and AUTT1 are set. Otherwise, discard the sample.
- Read both the AUX0 and AUX1 timestamp values and compare the values. Discard the values if they differ by more than the sampling uncertainty of 8 ns (if the SDP inputs are balanced externally) or slightly higher if the external trace lengths differ significantly.

Using this method, along with a software filter for expected values, almost all errors can be filtered out, with the remaining samples having a very high probability of being correct.

When using Port 0, the following combinations of SDP connections to AUX0 and AUX1 allow the above method to filter out all errors.

| **AUX0 Connection** | **AUX1 Connection** |
|---|---|
| SDP0 | SDP1 |
| SDP0 | SDP3 |
| SDP1 | SDP0 |
| SDP1 | SDP2 |
| SDP2 | SDP0 |
| SDP2 | SDP1 |
| SDP2 | SDP3 |
| SDP3 | SDP2 |

Status: No Fix.

## 36. EndPoint (EP): Spurious MSI/MSIX Interrupt Generated

Problem: When binding an interrupt from the default interrupt core (core 0) to the associated interrupt core, extra interrupts may be generated when clearing the PCI_MSIX_ENTRY_VECTOR_CTRL during an active interrupt session.

Implication: Extra interrupts may be generated during interrupt core binding.

Workaround: Device driver workaround has been identified and deployed to handle the extra interrupts.

Status: No Fix

## 37. EndPoint (EP): PCIe EP May not Detect Unexpected Completion Packets

Problem: The PCIe EP may drop packets with unexpected completions if the unexpected completion packets are received back-to-back.

Implication: Unexpected completion armored packets may not be detected and logged.

Workaround: None.

Status: No Fix.

## 38. GbE: GbE Near End Analog Loopback Not Supported

Problem: The GbE Near End Analog Loopback is not supported by the PCH GbE Interface.

Implication: GbE Near End Analog Loopback is not supported for Debug purposes.

Workaround: None.

Status: No Fix.

## 39. PCIe Root Port (RP): PCIe RP May not Automatically Switch into Compliance Mode

Problem: The PCIe RP may not automatically switch into compliance mode when the transmitter is terminated to 50-ohm test load.

Implication: Affects compliance testing.

Workaround: Software configuration may be used to force the link to enter compliance mode in both components of the link and then initiate a hot reset on the link.

Status: No Fix.

## 40. EndPoint (EP): PCIe EP Incorrect Completion ID

Problem: The Bus Number (B) section within a Completion ID (B:D:F) sent to the Host by the PCIe EP is always set to '0' for any completion response. The Device and Function Numbers (D:F) sections are set correctly.

Implication: Violation of PCIe Specification for Type 0 Configuration Write Requests where Functions are required to capture the Bus Number in the Completion ID of the response.

Workaround: None.

Status: No Fix.

## 41. EndPoint (EP): PCIe EP Link Status Register 2 (LNKSTS2) Not Implemented for Functions[1-4]

Problem: The PCIe LNKSTS2 Register in EP Functions[1-4] is not implemented. Reading the registers via Functions[1-4] returns indeterminate values

Implication: The LNKSTS2 Register cannot be read via Functions[1-4].

Workaround: Read the LNKSTS2 Register via Function 0.

Status: No Fix.

## 42. EndPoint (EP): PCIe Malformed Packet Checking is not implemented

Problem: A malformed PCIe TLP (Transaction Layer Packet) is a TLP that violates specific TLP formation rules as defined in the PCI Express Base Specification Revision 2.0 dated December 20, 2006. The PCIe EndPoint does not implement checks for malformed TLPs, and thus no malformed TLP errors will be reported by the device. The PCIe EndPoint does not transmit malformed TLPs.

Implication: Reception of the following types of malformed TLPs by the PCIe EndPoint may cause silent data corruption:

- The size of the data payload of a received TLP as given by the TLP's length field exceeds the length specified by the value in the Max_Payload_Size field of the PCIe EndPoint (Sec 2.2.2 of PCIe Base Spec).
- The value in the TLP's length field does not match the actual amount of data included in the TLP (Sec 2.2.2, Sec 2.2.9 of PCIe Base Spec).
- A TLP where the TLP Digest (TD) bit value does not correspond with the observed size (accounting for the data payload, if present), (Sec 2.2.3 of PCIe Base Spec).
- All received TLPs which use undefined Type field values (Sec 2.3 of PCIe Base Spec).

Workaround: As of October 2011 there were no published malformed errata/sightings that did not have a workaround for all Root Ports or Downstream ports listed below that would allow a malformed formed packet to be sent to the EndPoint:

PCIe EndPoint directly connected to any of the PCIe Root Ports in these platforms

- Sandy Bridge EN/EP + Patsburg PCH
- Sandy Bridge EN/EP + Chipset 89xx Series PCH
- Sandy Bridge Gladden + Chipset 89xx Series PCH
- Sandy Bridge Desktop + Cougar Point PCH
- Ivy Bridge EN/EP + Patsburg PCH
- Ivy Bridge Gladden + Chipset 89xx Series PCH
- Ivy Bridge Desktop + Panther Point PCH
- Jasper Forest + IbexPeak PCH
- Nehalem + Tylersburg IOH + ICH10
- Westmere + Tylersburg IOH + ICH10

PCIe EndPoint directly connected to a downstream port of the following PCIe Switches:

- PLX 87* family
- IDT IDT89HPES32NT24AG2 switch

Status: No Fix

## 43. EndPoint (EP): PCIe EP Virtual Functions Wrongly Reporting Advisory Errors as Correctable Errors

Problem: The EP wrongly report Advisory Non-Fatal errors as Correctable Errors (ERR_COR_Messages) instead of as Non-Fatal Errors (ERR_NONFATAL Messages).

Implication: System Software should handle these Advisory Errors as Non-Fatal Errors.

Workaround: None

Status: No Fix

## 44. EndPoint (EP): PCIe EP Sends Two Error Messages to the Host for IO_RD to Unsupported Address Space

Problem: When an unsupported request is detected with an IO_RD, the PCIe EP may return two error messages to the Host.

Implication: Induces multiple handling for the same error.

Workaround: Software should avoid IO_RD to unsupported address space

Status: No Fix

## 45. EndPoint (EP): PCIe EP May Generate Spurious Error Message

Problem: When clearing an Error status bit while handling an error, the PCIe Endpoint may spuriously generate another Error Message for the same error.

Implication: System Software may be invoked for an error that has already been handled.

Workaround: None

Status: No Fix

## 46. EndPoint (EP): PCIe EP Incorrect Handling of Multiple Errors

Problem: If a function has a pending error which has not yet been handled and cleared, and a second error is detected by another function, the EP does not report the second error.

Implication: If the second error is an uncorrectable error, it goes undetected and may affect system operation.

Workaround: None

Status: No Fix.

## 47. EndPoint (EP): PCIe EP Incorrect Handling of Poisoned Memory Writes.

Problem: The EP is supposed to handle a poisoned memory write as an Unsupported Request and abort the request, but the EP completes the write to the final destination with the poisoned data, and reports an error.

Implication: May cause unpredictable behavior if the writes target control structures.

Workaround: None

Status: No Fix

## 48. EndPoint (EP): PCIe EP Does Not Detect Poisoned Data for IO_WR Transactions

Problem: IO_WR transactions with poisoned data completes successfully without detection.

Implication: Target Registers may be corrupted.

Workaround: Software should perform a read after a write to verify the register value

Status: No Fix

## 49. EndPoint (EP): PCIe EP Does Not Reliably Update the Header Log and First Error Pointer

Problem: The Header Log and First Error Pointer are required to be updated for masked errors. If an unmasked error occurs after a masked error was detected, the Header Log and First Error Pointer are not be updated with the masked error information.

Implication: System Software is unable to determine the PCIe transaction that caused the error.

Workaround: None

Status: No Fix

## 50. EndPoint (EP): PCIe EP Does Not Report the Function Number Associated with a Poisoned TLP Error

Problem: The PCIe EP reports Poisoned TLP error using Function Number 0 regardless of the Function that received the poisoned transaction.

Implication: Wrong error source indicator for errors not associate with Function 0.

Workaround: System software must scan all the PCIe EP functions to identify the error

Status: No Fix

## 51. EndPoint (EP): PCIe EP Does Not Send Error Message for MMIO Read transaction to Unsupported Memory Space

Problem: The PCIe EP does not send an error message for MMIO Read access to unsupported memory space even though it indicates the error in the Error Status Register and completes the Read Request with Unsupported Request Status.

Implication: None

Workaround: System software should not access Unsupported Memory space

Status: No Fix

## 52. EndPoint (EP): Active State Power Management (ASPM) Not Supported

Problem: The PCIe EP does not support Active State Link Power Management.

Implication: If Active State Power Management is enabled and the link enters L0s, the EP may not re-enter L0. This will cause the system to hang.

Workaround: Disable Active State Power Management.

Status: No Fix

## 53. EndPoint(EP): Incorrect Logging & Reporting of Data Link Protocol Errors (DLPEs)

Problem: The PCIe Specification requires that non function-specific DLPEs should be logged by all functions in the device which are configured to log the error, and should be reported by by the configured functions. However, at the EP, only the last active function (last function which received a Config transaction) reports the error.

Implication: Violation of PCIe Specification for DLPE Error Reporting

Workaround: None

Status: No Fix

## 54. EndPoint (EP): PCIe EP Transition from D3Hot to L1

Problem: If a MEM_RD or IO_WR request is received when the EP is in D3Hot state with the Link in L1, the EP appropriately transitions to L0 and reponds to the request as Unsupported Request (UR). However, the link fails to transition back to the L1 state after the completion and stays in L0 state.

Implication: Violation of PCIe Power Management specifications.

Workaround: None

Status: No Fix

## 55. EndPoint (EP): Some PCIe EP Configuration Registers have Wrong Attribute Assignment

Problem: Some EP configuration registers have been wrongly assigned with Read/Write-Once-Sticky (RWOS) access attributes instead of Read-Only (RO) attributes.

The incorrect attribute affect the following Physical and Virtual Function (PF/VF) registers:

- PCC - PF Class Code Register
- PSVID - PF Subsystem Vendor ID
- PSID - PF Subsystem ID Register
- PLCAPR[0:3] - PF Link Capabilities Register
- PLSR[0:3] - PF Link Status Register
- PSRIOVFDID - PF SRIOV VF Device ID
- VCC[0:15] - VF Class Code Register
- VSID[0:15] - VF Subsystem ID Register

Implication: May fail PCIe Compliance tests.

Workaround: None

Status: No Fix

## 56. EndPoint (EP): PCIe EP Transition from D3hot to D0uninitialized

Problem: When the EP responds to a PCIe transition from D0uninitialized to D3hot, the link appropriately transitions from L0 to L1. However, when the EP is transitioned back from D3hot to D0uninitialized, the link transitions from L1-to-L0-to-L1, instead of L1-to-L0.

Implication: Violation of PCIe Power Management specifications.

Workaround: Perform the transition through D0active (D0unitialized-to-D0active-to-D3hot).

Status: No Fix

## 57. EndPoint (EP): PCIe EP Link Control Register 2 (PLCNTLR2) Sticky Bits Not Retaining value after Hot Reset

Problem: The EP PCIe Configuration space PLCNTLR2 Register (Configuration: Offset 0xA4) bits with Read-Write-Sticky (RWS) attributes are reset to ‘0b’ after Hot Reset.

Implication: Functions that require the bits to remain sticky are impacted. For example, it impacts system compliance testing.

Workaround: None.

Status: No Fix

## 58. USB: PLL Configuration Settings for USB-IO Interface

Problem: For PCH B0 Stepping, the power-up default clock phase may cause transmit clock timing violations.

Implication: The USB transmitter may not function properly if the appropriate clock phase is not selected.

Workaround: BIOS workaround has been identified and should be implemented as follows:

- BIOS should program USBIR<n>, Bit[31] = ‘1b’;
- BIOS should program PLLCTL2.U2TXPSEL (RCBA, Offset 0x356C), Bits[31:29] = 0x3.
- BIOS should program USBIR<n>, Bit[31] = ‘0b’

Status: No Fix

## 59. EndPoint (EP): PCIe Transaction Pending Bit Initialization

Problem: The Transaction Pending bit in the PCIe Status Register (PPDSTAT.TP) indicates that the PCIe Function has pending Non-Posted requests awaiting completions. The PCIe Specification requires the bit be cleared when all pending requests have completed successfully or completed with a timeout. At the EP, the bit is cleared only for transactions that complete successfully; it is not cleared if the pending transaction complete with a timeout.

Implication: The Transaction Pending bit is one of the resources that Software uses to quiesce a function prior to issuing a Function Level Reset (FLR)

Workaround: None.

Status: No Fix

## 60. EndPoint (EP): Incorrect IO Transaction Response in D3hot State

Problem: The EP responds normally to all IO transactions when in D3hot state, as if in D0 state, instead of responding with Unsurpported Request (UR).

Implication: Violation of PCIe Specifications.

Workaround: Avoid IO transactions when EP is in D3hot state.

Status: No Fix

## 61. PCH: IEEE Std. 1149.6 EXTEST_PULSE and EXTEST_TRAIN Instructions Not Supported by Non-EP_JTAG Port.

Problem: The PCH contains two separate JTAG (TAP) ports. One is used by the Non-Endpoint section on the PCH (JTAG) and the second is for the Endpoint section on the PCH (EP_JTAG). Both JTAG ports are compatible with the IEEE Std. 1149.1; but only the EP_JTAG is compatible with IEEE Std. 1149.6 instructions set. The Non-EP JTAG(JTAG), which connects to the DMI/SATA/PCIe (Root)/USB legacy I/O interfaces, is not compatible with the IEEE Std. 1149.6 and does not support the EXTEST_PULSE and EXTEST_TRAIN instructions.

Implication: IEEE Std.1149.6 instruction set not supported by the Non-EP JTAG port

Workaround: None.

Status: No Fix

## 62. PCH: Auxiliary (AUX) and Auxiliary2 (AUX2) Thermal Trip Interrupts not Functional.

Problem: The generation of PCI interrupts on AUX and AUX2 thermal trip points in the Thermal Sensors (TS0 & TS1) do not work properly.

Implication: Thermal interrupts are limited to Catastrophic and Hot state only.

Workaround: BIOS should disable the generation of interrupts on AUX and AUX2 thermal trips (TS0PIEN/TS1PIEN; TBARB + 0x82/0xC2) as follows:

- BIOS should program TS0PIEN/TS1PIEN, Bit[0,3,4,7] = ‘0b’.

Status: No Fix

## 63. EndPoint (EP): PCIe EP Does Not Report Timeout Errors

Problem: When the PCIe Endpoint transaction timer expires, an error message is not sent to the Host. However, all the relevant error status bits are updated correctly.

Implication: Results in incomplete transaction which may cause the device to hang.

Workaround: Device driver should implement a timeout mechanism to detect device hang condition. This timeout mechanism can then be used by the software to intervene and reset the device

Status: No Fix

## 64. EndPoint (EP): PCIe EP Does Not Register Correctable Errors

Problem: When the PCIe EP receives and detects a Receiver Error it does not set the Correctable Error Detected (CED) bit in the PF PCIe Device Status Register (PPDSTAT[0]). However, a correctable error message is sent to the Host and the Receiver Error bit (RES) is set in the PF PCIe AER Correctable Error Register (PPAERCS[0]).

Implication: Violation of system error reporting.

*Note:* Since the PCIe EP supports AER features, the device driver can use the AER Correctable Error Status register to identify the cause of the correctable error message.

Workaround: None

Status: No Fix

## 65. EndPoint (EP): PCIe EP Incorrect Handling of Read Completions with Unsupported Request (UR) Status

Problem: When the PCIe Endpoint receives a Read Completion with UR status it does not indicate the UR condition to the Requester of the read. The Requester may wait indefinitely for the completion of the transaction which can cause IO requests back-up at the device.

Implication: The transaction incompletion may cause the device and system to hang.

Workaround: Device driver should implement a timeout mechanism to detect device hang condition. This timeout mechanism can then be used by the software to intervene and reset the device.

Status: No Fix

## 66. EndPoint (EP): PCIe EP does not Enable the Correct number of Virtual Functions (VFs)

Problem: The NUMVF field of PF SRIOV Number of VFs Register (PSRIOVNUMVF), (BM:D0:F0; Offset 150h - 153h, Bits[15:0]) allows software to define the number of VFs assigned to a Physical Function (PF) as part of the process of creating VFs. Programming the NUMVF field does not enable the correct number of VFs as expected. The actual number of Enabled and Expected VFs as shown as follows:

| NUMVF | Actual # of VFs Enabled | Expected # of VFs |
|---|---|---|
| 0x00 | 0 | 0 |
| 0x01 | 2 | 1 |
| 0x02 | 3 | 2 |
| 0x03 | 4 | 3 |
| 0x04 | 5 | 4 |
| 0x05 | 6 | 5 |
| 0x06 | 7 | 6 |
| 0x07 | 8 | 7 |
| ---- | ---- | ---- |
| 0x0F | 16 | 15 |
| 0x10 | Undefined | 16 |

Implication: Inhibits the capability to enable only 1 VF. Provides capability to enable 2-16 VFs.

Workaround: None

Status: No Fix

## 67. GbE: SGMII Interface Receiver does not meet Input Differential Hysteresis (Vhyst) Specification

Problem: The SGMII Specification requires the receiver to ignore signals below 25 mV hysteresis voltage. However, the receiver may change states on input voltages close the 25 mV hysteresis specification which may cause unwanted state changes with non-monotonic input signals

Implication: Although this is a specification violation, it does not affect the functional operation of the interface.

Workaround: None.

Status: No Fix

## 68. SATA: Incorrect Number of Supported Ports Reported

Problem: The SATA AHCI Host Capabilities Register is configured to indicate that the SATA Controller supports 4 ports (B0:D31:F2: ABAR + 00h, Bit[4:0] = "03h"). The number of supported ports in the register (NPs) should have been set to "01h" to indicate support for 2 ports.

Implication: Incorrect reporting of supported ports. All PCH SKUs support only two ports.

Workaround: Driver should ignore the setting in the CAP Register, but instead to use the BIOS configuration of the "Active Ports" in the Ports Implemented Register (B0:D31:F2: ABAR + 0Ch, Bit[5:4]).

Status: No Fix

## 69. Endpoint (EP): PCIe EP Does not Indicate Poisoned Data from Reads of Corrupted Memory

Problem: The EP fails to set the Poisoned Data bit (EP bit) in the completion header for memory reads with corrupted data.

Implication: Minimal impact because IA software is notified via interrupts when internal errors are detected.

Workaround: None.

Status: No Fix

## 70. Endpoint (EP): PCIe EP Link May Not Train

Problem: The default mux setting selects the wrong analog detect signal which may cause the internal State Machines to get out of sync.

Implication: If the state machines get out of sync, the link will not train.

Workaround: This issue can be fixed with an EEPROM image upgrade.

- Using the Intel starter images:
  - The EEPROM Version = 3.0 or higher, includes the workaround.

OR

- If you build your own image with EICT, include "LAN0-CSR_Auto_Config_Power_Up_Ptr_0x27-CSR-Bottom_C.txt" version2 or higher in the EICT Pointer Field "LAN0 > LAN0 CSR Auto Config Power Up Ptr".

*Note: Jntel recommends that the latest EEPROM Version (version 3.0 or higher) should be used*

Status: No Fix

## 71. GbE: SGMII Interface Transmit Rise ($t_{rise}$) and Fall ($t_{fall}$) AC Timing Specification Violation

Problem: The SGMII interface transmit Rise ($t_{rise}$) and Fall ($t_{fall}$) times do not meet the SGMII AC Timing Specification

- The expected specification timing for $t_{rise}$ and $t_{fall}$ is 100 psec (min) and 200 psec (max) for 20% - 80% voltage rise or fall.
- The measured PCH SGMII interface timing is 85 psec for 20% - 80% voltage rise or fall.

Implication: Although this is a specification violation, it does not affect the functional operation of the interface.

Workaround: None.

Status: No Fix

## 72. QuickAssist Technology (QAT): Decompression Adler32 Checksum May be Calculated Incorrectly

Problem: Decompression checksum computed for Adler32 requests may produce erroneous results.

Implication: May cause valid decompression packets to be discarded.

Workaround: Software should be tasked to calculate the checksum for Adler32 requests.

*Note:* Refer to Issue# **IXA00378662** in the Intel® Communications Chipset 89xx Series Software - Release Notes (Doc# 441779) for Software solution.

Status: No Fix

## 73. Endpoint (EP): PCIe EP Link Disable requires Hot Reset

Problem: The EP does not perform an internal upstream port reset on PCIe Link Disable as required by the PCI Express specification.

Implication: If Link Disable is performed the credit consumed counters are not reset, once the link re-trains the device will hang due to lack of available credits.

Workaround: After clearing the Link Disable bit in the Root or Switch Downstream Port connected to the PCH, a Hot Reset must be performed by setting the Secondary Bus Reset bit of the Bridge Control Register associated with the same Root or Switch Downstream Port.

Status: No Fix

## 74. GbE: GbE MACs PCI Config Space Does not contain Valid VID/DID Default Values

Problem: The GbE MACs do not have valid VID/DID default values in the PCI configuration space for the PCH C0 & C1 Steppings.

Implication: If an EEPROM with a valid image is not loaded during power-up or system reset, then the GbE MACs VID and DID may be randomly set and invalid.

*Note:* It is a mandatory requirement to have a pre-programmed EEPROM image in the system; hence this is only an **ALERT** to provide the EEPROM.

Workaround: Ensure that the platform contains a pre-programmed EEPROM with valid VID/DID before powering up the system.

Status: No Fix

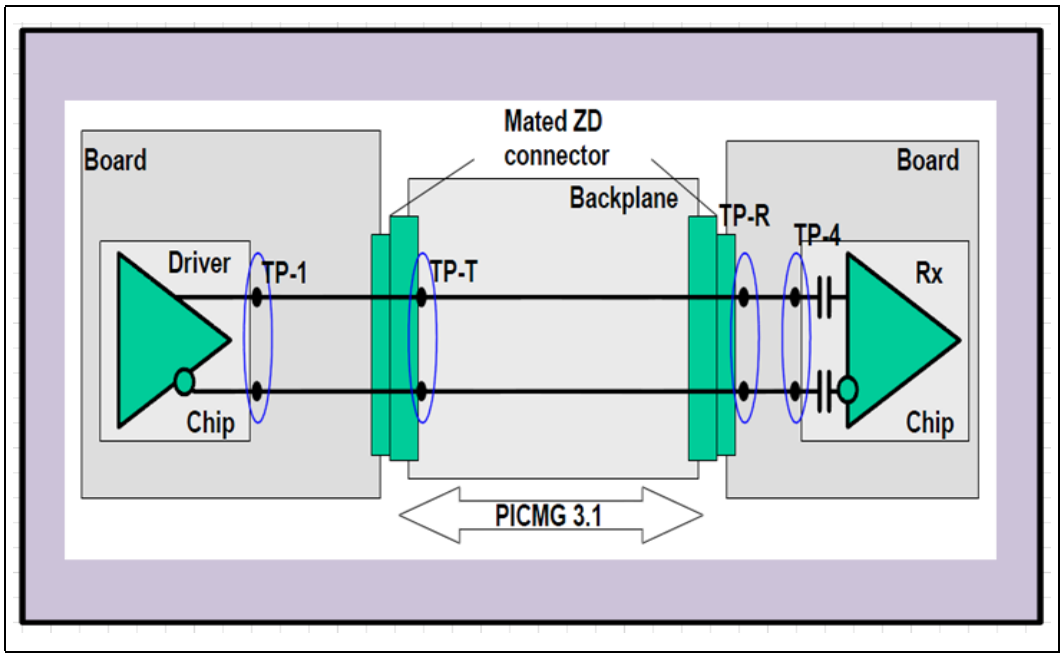
## 75. GbE: GbE Interface Transmit Voltage Level Specification Violation

Problem: The GbE 1000BASE-BX transmit buffers have been designed to maximize performance and robustness over a variety of routing scenarios. As a result, the transmitter signaling voltage levels may exceed the maximum motherboard TX connector and RX connector peak-to-peak voltage specification of 1350 mV by about 5%. (See Transmitter Specification at TP-T (Table 17), of PICMG 3.1, R1.0 Specification)

Implication: Although this is a specification violation, it does not affect the functional operation of the interface.

Workaround: Ensure that the signal routing length from TP-1 to TP-4 is greater than 10”. Refer to Intel® Communications Chipset 89xx Series Platform Design Guide (PDG) for maximum routing length. (See Figure 2).

**Figure 2. 1000BASE-BX/FC-PI Transmitter Electrical Specifications**




Status: No Fix

## 76. USB: USB Full/Low Speed Port Reset or Clear Transaction Translation (TT) Buffer Request

Problem: One or more full/low speed USB devices on the same RMH controller may be afftected if the devices are not suspended and either (a) software issues a Port Reset OR (b) software issues a Clear TT Buffer request to a port executing a split full/low Speed Asynchronous Out Command.

- The small window of exposure for Full-Speed device is around 1.5 micro-seconds and around 12 micro-seconds for Low-Speed device

Implication: The affected port may stall or receive stale data for a newly arrived split transfer occurring at the time of the Port Reset or Clear TT Buffer request.

***Note:*** This issue has only been observed in a synthetic test environment.

Workaround: None.

Status: No Fix.

## 77. USB: USB RMH Think Time Issue

Problem: The USB RMH Think Time may exceed its declared value in the RMH hub descriptor register value of 8 full-speed bit times.

Implication: If the OS USB driver fully subscribes to USB microframe, LS/FS transactions may exceed the microframe boundary.

***Note:*** No functional failures have been observed.

Workaround: None.

Status: No Fix.

## 78. USB: USB RMH False Disconnect Issue

Problem: The PCH may falsely detect a USB High-Speed (HS) device disconnect if all of the following conditions are met:

1. The HS device is connected through the Rate Matching Hub (RMH) of teh PCH's EHCI controller either directly or through a high-speed hub or series of high-speed hubs.
2. The device is resuming from selective suspend or port reset
3. The resume occurs within a narrow time window during the EOP (End of Packet) portion of the SOF (Start of Frame) Packet on the USB bus.

Implication: Following the false disconnect, the HS device will be automatically re-enumerated. The system implication will depend on the cause of the resume event:

- If the resume event is a port reset, a second port reset will be automatically generated and the device re-enumerated. No end user impact is expected.
- If the resume event is a hardware or software initiated resume from selective suspend, the implication will be device and software specific, which may result in anomalous system behavior..

***Note:*** If the HS device is a hub, then all of the devices behind the hub, independent of the device speed, may also be re-enumerated.

Workaround: None.

Status: No Fix.

# Specification Changes

There are no specification changes in this revision of the specification update.

# Specification Clarifications

## 1. GbE: Use of Wake on LAN Together with Manageability

Clarification: The Wakeup Filter Control Register (WUFC) contains the NoTCO bit, which affects the behavior of the wakeup functionality when manageability is in use. Note that if manageability is not enabled, the value of NoTCO has no effect.

When NoTCO contains the hardware default value of 0b, any received packet that matches the wakeup filters will wake the system. This could cause unintended wakeups in certain situations. For example, if Directed Exact Wakeup is used and the manageability shares the host's MAC address, IPMI packets that are intended for the BMC will wake the system, which might not be the intended behavior.

When NoTCO is set to 1b, any packet that passes the manageability filter, even if it also is copied to the host, is excluded from the wakeup logic. This solves the previous problem since IPMI packets will not wake the system. However, with NoTCO = 1b, broadcast packets, including broadcast magic packets, will not wake the system since they pass the manageability filters and are therefore excluded.

The Intel Windows* drivers set NoTCO by default.

| Effects of NoTCO Settings WoL | NoTCO | Shared MAC Address | Unicast Packet | Broadcast Packet |
|---|---|---|---|---|
| Magic Packet | 0b | - | OK | OK |
| Magic Packet | 1b | Y | No wake | No wake |
| Magic Packet | 1b | N | OK | No wake |
| Directed Exact | 0b | Y | Wake even if MNG packet. No way to talk to BMC without waking host. | N/A |
| Directed Exact | 0b | N | OK | N/A |
| Directed Exact | 1b | - | OK | N/A |

## 2. GbE SMBus: Illegal STOP Condition

Clarification: It is important to prevent illegal STOP conditions on the SMBus interface, even when resetting the Management Controller (MC).

Specifically, a STOP condition should never be generated by the MC during the high clock phase of an ACK cycle while reading packet data from GbE Controller as part of a Receive TCO LAN packet transaction.

If this situation occurs, the Controller replies with a NACK to all future commands until a power cycle. As a result, the SMBus interface becomes inoperable.

Workaround: Ensure that this illegal sequence does not occur, even during MC reset.

## 3. GbE SERDES: AN_TIMEOUT Only Works When Link Partner Idle

Clarification: The auto-negotiation timeout mechanism (PCS_LCTL.AN_TIMEOUT_EN) only works if the SerDes partner is sending idle code groups continuously for the duration of the timeout period, which is the usual case.

However, if the partner is transmitting packets, an auto-negotiation timeout will not occur since auto-negotiation is restarted at the beginning of each packet. If the partner has an application that indefinitely transmits data despite the lack of any response, it is possible that a link will not be established.

Workaround: If this is a concern, the auto-negotiation timeout mechanism may be considered unreliable and an additional software mechanism could be used to disable auto-negotiation if sync is maintained without a link being established (PCS_LSTS.SYNC_OK=1b and PCS_LSTS.LINK_OK=0b) for an extended period of time.

# Documentation Changes

There are no documentation changes in this revision of the specification update.

§ §