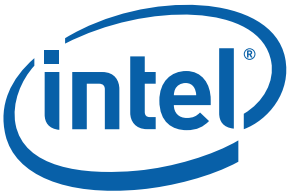


Intel® 82571EB/82572EI Ethernet Controller

Specification Update

January 2012
Revision 6.5

Legal Information

Revision History

Revision	Revision Date	Description
1.0	Jan 2004	Initial release.
1.1	Jan 2004	Added Erratum 16; Sightings 4-6.
1.2	Feb 2004	Added Erratum 17-18; Added Device Identification and Mechanicals
1.3	Mar 2004	Added B-1 errata.
1.4	Apr 2004	Added erratum 23 and 24. Added stepping information on summary of changes table.
1.5	May 2004	Removed references to 82570EI. Updated Erratum #14.
1.6	Jun 2004	Added erratum #30. Moved sightings #1, 2, 3, 6, 7, and 8 to erratum # 21, 25, 26, 27, 28, and 29, respectively. Removed sighing # 4.
1.7	Jul 2004	Fixed the summary of table changes. Appended updated schematics.
1.8	Jan 2005	Removed errata that were fixed for C0. Added new errata found on C0. Added 82572EI information.
1.9	Jun 2005	Removed errata fixed for C0. Added new errata for D0.
2.0	Sep 2005	Added errata 48 through 75, sightings 15 through 18 and specification clarifications 4, 5 & 6.
2.1	Feb 2006	Removed all pre-production completed/fixed items and renumbered the remaining items; added Specification Change 1; added Errata 35, 36 & 37; moved information about Specification Clarification 6—"On-Die Cable Discharge Event protection may not be sufficient" to design guide; changed Sighting 17 to Errata 37; removed Sighting 18: it was caused by a test setup issue; clarified some wording.
2.2	Jun 2006	Added Specification Change #2 (iSCSI Header Split Not Supported) and Change #3 (EEPROM Initialization). Added Errata 38-44. Added Specification Clarifications #6 & #7.
2.3	Nov 2006	Added Errata 45-56 and Specification Clarification 8.
3.0	March 2007	Added Errata 57-64 and Specification Clarification 9; corrected device names in MM number table; added alternative workaround for errata #7; added definition for "image" in table 2.
4.0	October 2007	Added Errata 65-67.
5.0	January 2008	Added Errata 68 and Specification Clarifications 10 & 11.
6.0	December 2008	Added Specification Change 4. Added Errata 69, 70, 71, 72 and 73; added Specification Clarifications 12 and 13: updated Errata 7, 18, and 66.
6.1	April 2009	Added Errata 74; added information to ECC Correction Enable in Specification 4;
6.2	June 2009	Added Specification Clarification 14.
6.3	July 2010	Added Errata:75-77; updated Errata 41, 68, and 69; added Specification Clarifications 15-16
6.4	January 2012	Added Specification Changes 5 and 6. Updated Errata 39 and 65. Added Errata 78 and 79. Added Specification Clarification 17. Added Software Clarification 1.
6.5	January 2012	Added Specifications Changes 7 and 8. Updated Errata 76. Updated Specification Clarification 16. Added SW Clarification 2.

1. Preface

This document applies to both the Intel® 82571EB and 82572EI Gigabit Ethernet controllers. They are commonly referred to as the device. Any information that applies to only one will be noted as such.

This document is an update to published specifications. Specification documents for this product include:

- *82571EB/82572EI Gigabit Ethernet Controller Datasheet*, Intel Corporation.
- *82571EB/82572EI Gigabit Ethernet Controller Design Guide*, Intel Corporation.
- *PCIe* Family of Gigabit Ethernet Controllers Software Developer's Manual,* Intel Corporation

This document is intended for hardware system manufacturers and software developers of applications, operating systems or tools. It may contain Specification Changes, Errata, and Specification Clarifications.

All product documents are subject to frequent revision, and new order numbers will apply. New documents may be added. Be sure you have the latest information before finalizing your design.

2. Nomenclature

This document uses various definitions, codes, and abbreviations to describe the Specification Changes, Errata, Sightings and/or Specification Clarifications that apply to the listed silicon/steppings:

Table 1. Definitions

Name	Description
Specification Changes	Modifications to the current published specifications. These changes will be incorporated in the next release of the specifications.
Errata	Design defects or errors. Errata may cause device behavior 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 on all devices.
Specification Clarifications	Greater detail or further highlights concerning a specification's impact to a complex design situation. These clarifications will be incorporated in the next release of the specifications.
Documentation Changes	Typos, errors, or omissions from the current published specifications. These changes will be incorporated in the next release of the specifications.

3. Codes and Abbreviations

Name	Description
X	Specification Change, Erratum, or Specification Clarification that applies to this stepping.
Doc	Document change or update that will be implemented.
Fix	This erratum is intended to be fixed in a future stepping of the component.
Fixed	This erratum has been previously fixed.
NoFix	There are no plans to fix this erratum.
Image	Erratum in the EEPROM image or one the can be fixed with an updated EEPROM image.
Eval	Plans to fix this erratum are under evaluation.
(No mark) or (Blank box)	This erratum is fixed in listed stepping or specification change does not apply to listed stepping.
Shaded	This Item is either new or modified from the previous version of the document.
DS	Data Sheet
DG	Design Guide
SDM	Software Developer's Manual
EDS	External Data Specification
AP	Application Note

4. Device Identification

The following tables and drawings describe the various identifying markings on each device package:

Table 2. Markings

Device	Stepping	Top Marking	Q-Specification	Notes
82571EB	D0 (lead free)	JL82571EB	Q866	Engineering Samples
82571EB	D0	HL82571EB	Q864	Engineering Samples
82572EI	D0 (lead free)	JL82572EI	Q867	Engineering Samples
82572EI	D0	HL82572EI	Q865	Engineering Samples
82571EB	D0 (lead free)	JL82571EB	N/A	Production
82571EB	D0	HL82571EB	N/A	Production
82572EI	D0 (lead free)	JL82572EI	N/A	Production
82572EI	D0	HL82572EI	N/A	Production
82571EB	D1	HL82571EB	N/A	Production
82571EB	D1 (lead free)	JL82571EB	N/A	Production
82571EI	D1	HL82571EI	N/A	Production
82571EI	D1 (lead free)	JL82571EI	N/A	Production
82571GB	D1 (lead free)	JL82571GB	N/A	Production
82571GI	D1 (lead free)	JL82571GI	N/A	Production

Note: The devices can also have a "82571GB" or "82572GI" marking (instead of "82571EB" or "82572EI"); the 82571GB and 82572GI devices are used only on Intel network interface adapters. The 82571GB is functionally equivalent to the 82571EB and the 82572GI is functionally equivalent to the 82572EI.

Table 3. Revision ID

Device	Vendor ID	Device ID	Revision ID*
82571EB D0/D1 (copper applications)	8086	105E	6
82571EB D0/D1 (fiber applications)	8086	105F	6
82571EB D0/D1 (SERDES backplane applications)	8086	1060	6
82572EI D0/D1 (copper applications)	8086	107D	6
82572EI D0/D1 (fiber applications)	8086	107E	6
82572EI D0/D1 (SERDES backplane applications)	8086	107F	6
*= Revision ID is located at Config address 0x8 bits 7:0			

Table 4. MM Numbers

Product	Tray MM#	Tape and Reel MM#
HL82571EB	875300 916297 (D1 stepping)	875296 916298 (D1 stepping)
HL82572EI	875302 916299 (D1 stepping)	875297 916300 (D1 stepping)
JL82571EB	875303 916663 (D1 stepping)	875298 916836 (D1 stepping)
JL82572EI	875304 916939 (D1 stepping)	875299 916940 (D1 stepping)
JL82571GB	875291 916846 (D1 stepping)	875275 916938 (D1 stepping)
JL82571GI	875293 916941 (D1 stepping)	875276 916946 (D1 stepping)

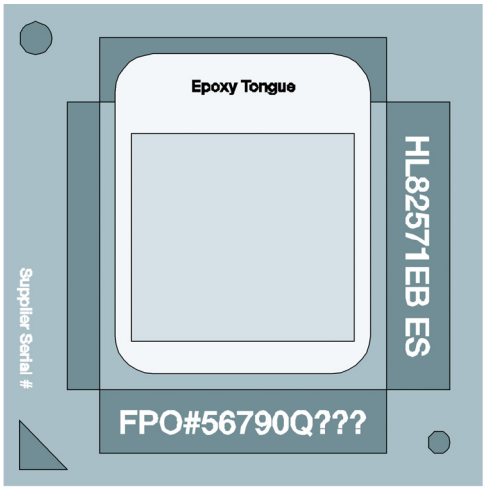


Figure 1. Example 82571EB/82572EI Identifying Marks

Note: Lead-free parts will have "JL" as the prefix for the product code (vs. "HL") and the "Q" designator refers to the Q Specification number in the table above.

Note: The devices can also have a "82571GB" or "82572GI" marking (instead of "82571EB" or "82572EI"); the 82571GB and 82572GI devices are used only on Intel network interface adapters. The 82571GB is functionally equivalent to the 82571EB and the 82572GI is functionally equivalent to the 82572EI.

Note: There is no change for parts listed as D1. There are no Form, Fit or Function changes to this silicon. Intel anticipates no impact to customers. This is an internal package change to provide a material solution that is RoHS compliant; Intel qualified and certified this change in the same way as it does for all products supplied to customers.

5. Summary Table of Changes

No.	D0	Plans	Specification Changes	Page
1	X	Doc	SMBus Operation at 1MHz Not Supported.	16
2	X	Doc	iSCSI Header Split Is Not Supported	16
3	X	Doc	The EEPROM Initialization Control 2 (word 0Fh) bit 7 is Reserved and Must Be Set To 0.	18
4	X	Doc	82571EB ECC Protection Enable 0x1100	18
5	X	Doc	PBA Number Format	18
6	X	Doc	Updates to PXE/iSCSI EEPROM Words	19
7	X	Doc	Using TCP Segmentation Offload with IPv6	20
8	X	Doc	Update Definition of SW EEPROM Port Identification LED Blinking (Word 0x4)	20
No.	**D0**	**Plans**	**Errata**	**Page**
1	X	NoFix	When Two Functions Have Differing MAX_PAYLOAD_SIZE, the Device Might Use the Larger Value For All Functions.	22
2	X	NoFix	Upstream Attempt to Reconfigure the PCIe Link by Moving the Link Training Status State Machine (LTSSM) from Recovery To Configuration Will Cause a "Link Down" Event.	22
3	X	NoFix	When Using Serial Over LAN, the Device's Power State Can Be Ambiguous.	22
4	X	NoFix	PCIe Differential- and Common-Mode Return Loss Is Higher than Specified Value.	23
5	X	NoFix	SerDes Transmit Differential Return Loss Is Higher than Specified Value.	23
6	X	NoFix	SerDes Is Unable To Acquire Sync From Ordered Sets Beginning With /K28.1/.	23
7	X	NoFix	Device Transmit Operation Might Halt in TCP Segmentation Offload (TSO) Mode when Multiple Requests Are Enabled.	24
8	X	NoFix	IDE-Redirect Persistent Retransmission Inconsistency.	24
9	X	NoFix	SMBus Transactions Might Be NACKed (Not ACKnowledged) Under IDE and SMBus Stress.	24
10	X	NoFix	I2C Transactions: When Working with Bus Speed 400KHz or Higher, Bus Might Hang when the Master Reads More Bytes than the Slave Reported.	25

11	X	NoFix	SOL Timeout Character Control Byte In EEPROM Image Does Not Function.	25
12	X	NoFix	Incorrect Number of Retransmissions of Link-Down Alert.	25
13	X	NoFix	Device Does Not Support PCIe Active State Power Management L1 State (ASPM L1).	26
14	X	NoFix	XOFF from Partner Can Prevent Flow-Control (XON/XOFF) Transmission.	26
15	X	NoFix	Missed RX Packets.	27
16	X	NoFix	Tx Stops during Host Management Stress in 10Mbps Half-Duplex.	27
17	X	NoFix	Device Overwrites Port A LAA to Default Value Due to Port B Software Reset.	28
18	X	NoFix	Enabling or Disabling RSS in the Middle of Received Packets May Stop Receive Flow.	31
19	X	NoFix	Packets with IPV6-Tunneled-in-IPV4 with a Certain Value Of Last IP Options Will Have an Incorrect RSS Hash Value.	31
20	X	NoFix	Formed and Invalid /C/ Code Handling on the SerDes Interface.	31
21	X	NoFix	False Detection of idle_match Condition on the SerDes Interface.	32
22	X	NoFix	Ability Match and Acknowledge Match on the SerDes Interface.	32
23	X	NoFix	Frames with Alignment Errors.	33
24	X	NoFix	Inter-Frame Spacing (10/100 Half-Duplex Mode).	33
25	X	NoFix	Auto Cross Sample Timer (PHY-related issue).	33
26	X	NoFix	Firmware Reset Occurs when Performing Transactions with a Low Interpacket Gap (IPG) Using Fast Management Link (FML) at 8MHz.	33
27	X	NoFix	10-base TLink Pulse Hits the Template Mask Due to Voltage Ripple/Glitch.	34
28	X	NoFix	10base-T TP_IDL Template Failure.	35
29	X	NoFix	BMC Fragments that Are Sent Through Two Different SMBus Ports Are Sent Over LAN as a Single Packet.	35
30	X	NoFix	Frames with Variations in the Preamble Are Rejected.	35
31	X	NoFix	Reception of Undersized Frames Affects Good Frame Reception.	36
32	X	NoFix	Packet Length-Related Issues.	36
33	X	NoFix	When MANC.EN_XSUM_FILTER Is Not Enabled, Received Packets With Wrong UDP Checksum Are Transferred To BMC.	36
34	X	NoFix	Device Sends Only One XOFF Even if the Link Partner Has Timed Out and It Is Still Congested.	37
35	X	NoFix	When Wake on LAN (WoL) Is Disabled, the Device Consumes More Than the Specified 20mA.	37
36	X	NoFix	The Device Does Not Correctly Handle Received Nullified Transaction Layer Packets (TLP).	37
37	X	NoFix	Link Down During Receive Flow May Cause Data Corruption.	38
38	X	Fixed	Incorrect PCIe Configurations Can Be Set by Earlier Versions of dev_starter EEPROM Images (v5.8 and below).	38
39	X	NoFix	Packets Received with an L2+L3 Header Length Greater than 256 Bytes Can Incorrectly Report a Checksum Error.	38

40	X	NoFix	PCIe Bus Can Halt upon D3/L1 Entry If There Are Less Than 16 Posted Data (PD) Flow Control Credits (=256byte memory writes).	39
41	X	Fixed	When APM Enable (WOL) is not set may affect the PCIe Configurations (dev_starter images v5.9 and below).	39
42	X	Fixed	Traffic on SMBus While Link Is Down Causes Firmware Reset.	40
43	X	NoFix	SOL Stress Data Integrity Fails with IDER Stress.	40
44	X	NoFix	The 82571EB/82572EI PCIe Transmit Differential Voltage Amplitude Is 1.4V (Maximum of 1.5V) for the First 15ms of Transmission.	40
45	X	Image	ARC Halts when SMBus Sslave Address is Set to 0x00.	40
46	X	Image	Rx Packet NotificationTimeout Does Not Reset after Master Reads Fragment.	41
47	X	Image	BMC Configuration Commands are Discarded when there is Heavy Manageability Traffic Load.	41
48	X	Image	Duplicate Fragments Might Be Sent to the BMC.	41
49	X	Image	Memory Buffer Leaks Under Heavy SMBus Traffic Load.	41
50	X	Image	First Two Bytes of a Rx Packet Forwarded to the BMC Might Be Dropped, Degrading Performance.	42
51	X	No Fix.	SMBus Might Hang if the BMC Is Reset in the Middle of a Transaction.	42
52	X	No Fix.	Certain Malformed IPV6 Extension Headers are not Processed Correctly by the Device.	42
53	X	No Fix	Completion with CA or UR Status Is Considered Malformed.	43
54	X	Image	HMAC Calculation For RMCP+ Session Establishment Is Incorrect.	43
55	X	Image	SOL Payload Fails to Activate with Encryption Activation Bit Set When Session Was Not Established with Encryption.	43
56	X	Image	User Password Not Being Used (Instead of the Kg) when Calculating the SIK.	43
57	X	Image	Firmware Resets While Link Is Down	44
58	X	Image	Integrity Value in RMCP+ session establishment	44
59	X	Image	Username in RAKP1 Message Must Be Padded to 16 Bytes	44
60	X	Image	Device Accepts Invalid User Name when RMCP+ Session Owner	45
61	X	Image	Configuring RMCP+ Password from the BMC	45
62	X	Image	"Update User Password" Command Incorrectly Accepts Less Than 20 Bytes of Data	45
63	X	No Fix	Byte Enables 2 and 3 Are Not Set on MSI Writes	45
64	X	No fix	Wakeup Event Occurs on Magic Packet that Doesn't Pass Address Filter	46
65	X	No Fix	SKP (SKIP ordered set) Will Reset TS (Training Sequence) Count.	46
66	X	No Fix	82571EB-82572EI Does Not Correctly Implement Master/Slave Resolution.	46
67	X	No Fix	82571EB-82572EI Improperly Implements the Auto-Negotiation Advertisement Register.	47
68	X	No Fix	PPCIe: Reception of Completion That Should Be Dropped May Occasionally Result In Device Hang or Data Corruption	47

69	X	No Fix	RDTR: No Write Back on RX Packet When Too Close to Previous Packets.	48
70	X	No Fix	82571/82572 Overwrites Transmit Descriptors in Internal Buffer.	48
71	X	No Fix	Link Indication: LED Remains On In D3 Power State in SerDes Mode.	49
72	X	No Fix	PCIe: Missing Replay Due to Recovery During TLP Transmission	49
73	X	No Fix	PCIe: LTSSM Moves from L0 to Recovery Only When Receiving TS1/TS2 on All Lanes	49
74	X	No Fix	Missing Interrupt Following ICR Read	49
75	X	No Fix	Tx Packet Lost After PHY Speed Change Using Auto-negotiation	50
76	X	No Fix	Tx Data Corruption When Using TCP Segmentation Offload	50
77	X	No Fix	PCIe: Extended PCIe "Hot Reset" Can Lead to a Firmware Hang	51
78	X	No Fix	SerDes: RXCW.Rx ConfigInvalid Set Incorrectly	51
79	X	No Fix	PCIe: Spurious SDP/STP Causes Packets to be Dropped	51
No.	**D0**	**Plans**	**Specification Clarifications**	
1	X	Doc	Disable Auto MDI-X for Forced 100BASE-TX Operation.Disable Auto MDI-X for Forced 100BASE-TX Operation.	53
2	X	Doc	Request Will Not Be Treated As Completion Abort (CA) When the Programming Model Bytes Enable Is Violated.	53
3	X	Doc	System-Level EMI Test Can Be Affected by 490MHz Harmonic Seen In 10Base-T Waveform Spectrum.	53
4	X	Doc	MDI Return Loss Is Marginal Near 40MHz at 115ohm Load.	53
5	X	Doc	PCIe Output Driver Amplitude Can Be Set Incorrectly By the EEPROM.	54
6	X	Doc	Only One Port Can Be Disabled at a Time; LAN Disable (LAN0_DIS_N & LAN1_DIS_N)—82571 Only	55
7	X	Doc	Manageability Modes Not Available When System Is in S5 State when "device power down" Is Activated and APM Is Disabled.	55
8	X	Doc	Manageability Not Supported on SMBus 1.	55
9	X	Doc	Support for WOL Concurrently on Both Ports	55
10	X	Doc	LED Modes Based On LINK Speed Only Work in Copper(Internal PHY) Mode	56
11	X	Doc	THERM_Dp (D4) and THEMR_Dn (D5) are reserved and should not be used	56
12	X	Doc	TCP Segmentation Offload Operations With Both Transmit Queues Enabled.	56
13	X	Doc	When Port 0 and Port 1 Are Connected Back-to-Back, the PHY Should Be Reset As Part of the Driver Initialization To Avoid Link Failures.	56
14	X	Doc	PCIe: Completion Timeout Mechanism Compliance	56
15	X	Doc	Critical Session (Keep PHY Link Up) Mode Does Not Block All PHY Resets Caused by PCIe Resets	57
16	X	Doc	Receiver Detection Circuit Design and Established Link Width.	57

17	X	Doc	Use of Wake on LAN Together with Manageability	58
No.	**D0**	**Plans**	**Software Clarifications**	
1	X	Doc	While In TCP Segmentation Offload, Buffers Limited to 64 KB	59
2	X	Doc	Serial Interfaces Programmed By Bit Banging	59

6. Specification Changes

1. SMBus Operation at 1 Mhz Is Not Supported (400 kHz Operation Not Affected)

Operation of the SMBus at 1 MHz is not supported. Operation at the standard SMBus frequency (400 kHz) is not affected and is supported. The operation frequency is set by the EEPROM.

2. iSCSI Header Split Feature Is Not Supported

The extended Rx and Rx write-back descriptors are affected. This information supercedes the information in the *PCIe Family of Gigabit Ethernet Controllers Software Developer's Manual*, Section 3.2.6.5.

The following tables reflect the changes:

PKTTYPE (bit 19:16):

The PKTTYPE field defines the type of the packet that was detected by the device. It tries to find the most complex match until it locates the most common one, as shown in the Packet Type table below:

Packet Type	Description
0x0	MAC , (VLAN/SNAP) Payload
0x1	MAC, (VLAN/SNAP) Ipv4 , Payload
0x2	MAC, (VLAN/SNAP) Ipv4, TCP/UDP, payload
0x3	MAC (VLAN/SNAP) ,Ipv4, Ipv6, payload
0x4	MAC (VLAN/SNAP) ,Ipv4, Ipv6 ,TCP/UDP, payload
0x5	MAC (VLAN/SNAP) , Ipv6, payload
0x6	MAC (VLAN/SNAP) , Ipv6 ,TCP/UDP, payload
0x8	MAC, (VLAN/SNAP) Ipv4, TCP/UDP, NFS, payload
0xA	MAC (VLAN/SNAP) ,Ipv4, Ipv6 ,TCP/UDP,NFS, payload
0xC	MAC (VLAN/SNAP) , Ipv6 ,TCP/UDP, NFS, payload

Note: Payload does not mean raw data, but can be also an unsupported header.
Note: If there is NFS header in the packets, it can be seen in the packet type field.

Packet types supported by the packet split: The 82571/82572 provides header split for the packet types listed below. Other packet types are posted sequentially in the buffers of the packet split receive buffers.

Packet Type	Description	Header Split
0x0	MAC , (VLAN/SNAP) Payload	No
0x1	MAC, (VLAN/SNAP) Ipv4 , Payload	Split header after L3 if fragmented packets
0x2	MAC, (VLAN/SNAP) Ipv4, TCP/UDP, payload	Split header after L4 if not fragmented, otherwise treat as packet type 1
0x3	MAC (VLAN/SNAP) ,Ipv4, Ipv6, payload	Split header after L3 if either Ipv4 or Ipv6 indicates a fragmented packet
0x4	MAC (VLAN/SNAP) ,Ipv4, Ipv6 ,TCP/UDP, payload	Split header after L4 if Ipv4 not fragmented and if Ipv6 does not include fragment extension header, otherwise treat as packet type 3
0x5	MAC (VLAN/SNAP) , Ipv6, payload	Split header after L3 if fragmented packets
0x6	MAC (VLAN/SNAP) , Ipv6 ,TCP/UDP, payload	Split header after L4 if Ipv6 does not include fragment extension header, otherwise treat as packet type 5
0x8	MAC, (VLAN/SNAP) Ipv4, TCP/UDP, NFS, payload	Split header after L5 if not fragmented, otherwise treat as packet type 1
0xA	MAC (VLAN/SNAP) ,Ipv4, Ipv6 ,TCP/UDP,NFS, payload	Split header after L5 if Ipv4 not fragmented and if Ipv6 does not include fragment extension header, otherwise treat as packet type 3
0xC	MAC (VLAN/SNAP) , Ipv6 ,TCP/UDP, NFS, payload	Split header after L5 if Ipv6 does not include fragment extension header, otherwise treat as packet type 5

As a result of this specification change, bits 5:0 of the Receive Filter Control Register are now reserved.

Field	Bit(s)	Initial Value	Description
Reserved	5:0	0	**Reserved**. Should be written with 0 to ensure future compatibility.
NFSW_DIS	6	0	NFS Write disable Disable filtering of NFS write request headers.
NFSR_DIS	7	0	NFS Read disable Disable filtering of NFS read reply headers.
NFS_VER	9:8	00	NFS Version 00 NFS version 2 01 NFS version 3 10 NFS version 4 11 Reserved for future use
IPv6_dis	10	0	IPv6 disable. Disable IPv6 packet filtering
IP6Xsum_dis	11	0	IPv6 Xsum disable Disable XSUM on IPv6 packets
ACKDIS	12	0	ACK accelerate disable When this bit is set OPHIR will not accelerate interrupt on TCP ACK packets.
ACKD_DIS	13	0	ACK data Disable 1 – OPHIR will recognize ACK packets according to the ACK bit in the TCP header + No –CP data 0 - OPHIR will recognize ACK packets according to the ACK bit only. This bit is relevant only if the ACKDIS bit is not set.
Field	**Bit(s)**	**Initial Value**	**Description**
IPFRSP_DIS	14	0	IP Fragment Split Disable When this bit is set the header of IP fragmented packets will not be set.
EXSTEN	15	0	Extended status Enable, When the EXSTEN bit is set or when the Packet Split receive descriptor is used, OPHIR writes the extended status to the Rx descriptor.
IPv6_ExdIS	16	0	**IPv6 Extension Header Disable,** Chicken bit to disable the IPv6 extension headers parsing for XSUM offload, Header split and Filtering: 0 – parse and recognize allowed IPV6 extension headers (Hop-by-Hop, Destination Options, and Routing) 1 – do not recognize above extension headers

NEW_IPV6_EXT_DIs	17	0	**New IPv6 Extension Header**, Chicken bit to disable the mobility IPv6 extension headers parsing, required for RSS: 0 – parse and recognize IPV6 "home address option" and "rout2" extension headers for RSS function 1 – If an IPv6 packet includes either a Home-Address-Option or a Routing-Header-Type-2, then the TcpIPv6Ex and IPv6Ex functions are not used.

3. The EEPROM Initialization Control 2 (word 0Fh) bit 7 is Reserved and Must Be Set To 0.

The EEPROM Initialization Control 2 (word 0Fh) bit 7 is reserved and must be set to 0.

Documents affected by this change are the *PCIe Family of Ethernet Gigabit Controllers Software Developers Manual* and the *82571EB/82572EI EEPROM Information Guide*.

4. 82571EB ECC Protection Enable 0x1100

The Packet Buffer has ECC protection and uses a register at address 0x1100 to control the operation of the ECC:

Field	Bit(s)	Initial Value	Description
ECC Error Counter	31:20	0x000	Incremented on each ECC error detection, cleared by writing to bit 1 of this register.
Reserved	19	1	Reserved write to 1
ECC Disable From EEPROM	18	Loaded from EEPROM word 0x10/0x20 bit 5	Read-only. Loaded from EEPROM. When set, disables ECC generation and error correction.
ECC CSR access enable	17	1	When set, enable ECC generation and error correction on CSR access to the Packet Buffer.
Reserved	16	1	Reserved. Write to 1
ECC error address	15:4	0xFFF	Contains the Packet Buffer address of the most recent ECC error. Out of reset this is set 0xfff (invalid value) Also set to 0xfff by writing to bit 1 of this register. This field is for Debug only.
Reserved	3	0	Reserved. Write to 0
ECC interrupt enable	2	0	When set, enables the setting of ICR bit 5 on detection of an ECC error
ECC Statistic Clear	1	0	Writing 1 to this bit clears the error counter and error address fields
ECC Correction Enable	0	0	When set, enables single bit ECC error correction. When clear, ECC errors will be detected, but not corrected. **Intel recommends that this bit be enabled**.

The 82571-82572 supports 256 KB TCP packets; however, each buffer is limited to 64 KB since the data length field in the transmit descriptor is only 16 bits. This restriction increases driver implementation complexity if the operating system passes down a scatter/gather element greater than 64KB in length. This can be avoided by limiting the offload size to 64 KB.

5. Updates to PBA Number EEPROM Word Format.

Change: PBA Number Module — Word 0x8-0x9

The nine-digit Printed Board Assembly (PBA) number used for Intel manufactured Network Interface Cards (NICs) is stored in EEPROM.

Through the course of hardware ECOs, the suffix field is incremented. The purpose of this information is to enable customer support (or any user) to identify the revision level of a product.

Network driver software should not rely on this field to identify the product or its capabilities.

PBA numbers have exceeded the length that can be stored as HEX values in two words. For newer NICs, the high word in the PBA Number Module is a flag (0xFAFA) indicating that the actual PBA is stored in a separate PBA block. The low word is a pointer to the starting word of the PBA block.

The following shows the format of the PBA Number Module field for new products.

PBA Number	Word 0x8	Word 0x9
G23456-003	FAFA	Pointer to PBA Block

The following provides the format of the PBA block; pointed to by word 0x9 above:

Word Offset	Description
0x0	Length in words of the PBA Block (default is 0x6)
0x1 ... 0x5	PBA Number stored in hexadecimal ASCII values.

The new PBA block contains the complete PBA number and includes the dash and the first digit of the 3-digit suffix which were not included previously. Each digit is represented by its hexadecimal-ASCII values.

The following shows an example PBA number (in the new style):

PBA Number	Word Offset 0	Word Offset 1	Word Offset 2	Word Offset 3	Word Offset 4	Word Offset 5
G23456-003	0006	4732	3334	3536	2D30	3033
	Specifies 6 words	G2	34	56	-0	03

Older NICs have PBA numbers starting with [A,B,C,D,E] and are stored directly in words 0x8-0x9. The dash in the PBA number is not stored; nor is the first digit of the 3-digit suffix (the first digit is always 0b for older products).

The following example shows a PBA number stored in the PBA Number Module field (in the old style):

PBA Number	Byte 1	Byte 2	Byte 3	Byte 4
E23456-003	E2	34	56	03

6. Updates to PXE/iSCSI Words

Gigabit Main Setup Options Word 0x30, 0x34. See following table.

Bit(s)	Value	Port Status	CLP(Combo) Executes	iSCSI Boot Option ROM CTRL-D Menu	FCoE Boot Option ROM CTRL-D Menu
2:0	0	PXE	PXE	Displays port as PXE. Allows changing to Boot Disabled, iSCSI Primary or Secondary	Displays port as PXE. Allows changing to Boot Disabled, FCoE enabled
	1	Boot Disabled	NONE	Displays port as Disabled. Allows changing to iSCSI Primary/ Secondary	Displays port as Disabled. Allows changing to FCoE enabled
	2	iSCSI Primary	iSCSI	Displays port as iSCSI Primary. Allows changing to Boot Disabled, iSCSI Secondary	Displays port as iSCSI. Allows changing to Boot Disabled, FCoE enabled
	3	iSCSI Secondary	iSCSI	Displays port as iSCSI Secondary. Allows changing to Boot Disabled, iSCSI Primary	Displays port as iSCSI. Allows changing to Boot Disabled, FCoE enabled
	4	FCoE	FCOE	Displays port as FCoE. Allows changing to port to Boot Disabled, iSCSI Primary or Secondary	Displays port as FCoE. Allows changing to Boot Disabled
	5-7	Reserved	Same as Disabled	Same as Disabled	Same as Disabled
4:3	Same as before	--	--	--	--
5	Bit 5, formerly used to indicate iSCSI enable/ disable, is no longer valid and is not checked by software.	--	--	--	--
15:7	Same as before	--	--	--	--

7. Using TCP Segmentation Offload with IPv6

When using TCP Segmentation Offload of IPv6 packets with two transmit queues, the following settings must be used:

- Program IPCSO equal to TUCSO in the context descriptor.
- Set IXSM in addition to TXSM in the data descriptor(s).

Intel Windows and Linux drivers (e1000e) only use one transmit queue for this device.

8. Update Definition of SW EEPROM Port Identification LED Blinking (Word 0x4)

Driver software provides a method to identify an external port on a system through a command that causes the LEDs to blink. Based on the setting in word 0x4, the LED drivers should blink between STATE1 and STATE2 when a port identification command is issued.

When word 0x4 is equal to 0xFFFF or 0x0000, the blinking behavior reverts to a default.

Bit	Description
15:12	Conrol for LED 3 0000b or 1111b: Default LED blinking operation is used. 0010b = Default in STATE1 + LED is ON in STATE2. 0011b = Default in STATE1 + LED is OFF in STATE2. 0100b = LED is ON in STATE1 + Default in STATE2. 0101b = LED is ON in STATE1 + LED is ON in STATE2. 0110b = LED is ON in STATE1 + LED is OFF in STATE2. 0111b = LED is OFF in STATE1 + Default in STATE2. 1000b = LED is OFF in STATE1 + LED is ON in STATE2. 1001b = LED is OFF in STATE1 + LED is OFF in STATE2. All other values are reserved.
11:8	Control for LED 2 - same encoding as for LED 3.
7:4	Control for LED 1 - same encoding as for LED 3.
3:0	Control for LED 0 - same encoding as for LED 3.

7. Errata

1. When Two Functions Have Differing MAX_PAYLOAD_SIZE, the Device Might Use the Larger Value For All Functions.

Problem: MAX_PAYLOAD_SIZE is programmed per function. If two PCIe functions have different MAX_PAYLOAD_SIZE, the device might use the larger value for all functions. The usage model for the device is to have all functions use the same MAX_PAYLOAD_SIZE.

Implication: There is no impact on the functional flow.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

2. Upstream Attempt to Reconfigure the PCIe Link By Moving the Link Training Status State Machine (LTSSM) From Recovery to Configuration Will Cause a "Link Down" Event.

Problem: The device will not move its LTSSM from Recovery to Configuration when it receives Training Sequences (TS) with only "lane number" set to PAD.

Implication: If the upstream component tries to reconfigure the link by moving the LTSSM from the Recovery.Idle state to the Configuration state (sending TS1s with only "lane number" set to PAD), the link will fail and the units will go to Detect states, causing a "link down" event.

Workaround: The upstream component should not apply this option.

Status: No Fix: There are no plans to fix this erratum.

3. When Using Serial Over LAN, the Device's Power State Can Be Ambiguous.

Problem: The same physical line is allocated for SMBus Alert and for S0 Power Indication. In Serial-Over-LAN (SOL), both are needed by manageability firmware, which treats these indications as separate. For LOMs containing SOL, the line is used for SMBus Alert.

Implication: There are two implications:

- SOL behavior might be confused because an SMBus Alert might be considered as a power state indication
- SOL cannot ascertain when a power state change has occurred

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

4. PCIe Differential- and Common-Mode Return Loss Is Higher Than Specified Value.

Problem: The PCIe transmitter's differential return loss is up to -9 dB instead of the -10 dB requirement. A PCIe Engineering Change Notice sets -10 dB as the requirement instead of the previous -15 dB in the base 1.0a specification..

The PCIe receiver's worst-measured differential return loss is up to -7.7dB instead of the -10dB requirement.

Implication: The out-of-specification return loss adds noise to the TX transmission line.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

5. SerDes Transmit Differential Return Loss Is Higher Than Specified Value.

Problem: The SerDes transmitter's differential return loss is up to -9 dB instead of the -10 dB requirement. A PCIe Engineering Change Notice sets -10 dB as the requirement instead of the previous -15 dB in the base 1.0a specification..

Implication: The out-of-specification return loss adds noise to the TX transmission line.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

6. SerDes Is Unable to Acquire Sync from Ordered Sets Beginning with /K28.1/.

Problem: Device SerDes is unable to acquire sync from ordered sets beginning with /K28.1/. If the link partner did not transmit any other characters that contain "commas" other than /K28.1/, the device will not attain sync.

Implication: Artificial testing of this portion of the standard will produce failures.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

7. Device Transmit Operation Might Halt in TCP Segmentation Offload (TSO) Mode when Multiple Requests (MULR) Are Enabled.

Problem The Device Transmit flow stops and the device hangs when operating in TSO with MULR enabled.

Implication: When operating in TCP Segmentation Offload mode and with Multiple Request enabled, one of the two workarounds listed below must be in place, or the Transmit Flow will stop unexpectedly.

Workaround: The driver must ensure that the first descriptor points to the (L2+L3+L4) Header and at least two bytes of the data (payload). This has been implemented in the Intel drivers. This workaround must be applied before activating TSO when MULR=1.

Alternatively, register 0x3940 "TARC1" bit 22 can be set at initialization time to workaround this issue.

Status: No Fix: There are no plans to fix this erratum.

8. IDE-Redirect Persistent Retransmission Inconsistency.

Problem: When sending the "StartIDERedirection" message from a remote management console, a "NumRetransmits" value of zero should define a persistent retransmission of "StartIDERedirection" messages until link is achieved. The device transmits only one "StartIDERedirection" message.

Implication: Using the value of zero is equivalent to using the value of one.

Workaround**:** Use a numeric value of NumRetransmits that is not zero or one.

Status: No Fix: There are no plans to fix this erratum.

9. SMBus Transactions Might Be NACKed (Not ACKnowledged) under IDE and SMBus Stress.

Problem**:** IDE and SMBus stress might cause a small percentage (<0.05%) of SMBus transactions to be NACKed. This is due to speed and memory limitations.

Implication**:** NACKing SMBus transactions does not impact function.

Workaround**:** Not applicable.

Status: No Fix: There are no plans to fix this erratum.

10. I²C Transactions: When Working with Bus Speeds 400 KHz or Higher, the Bus Might Hang When the Master Reads More Bytes than the Slave Reported.

Problem**:** When working in I²C mode, and when BMC executes an I²C read transaction, the device responds with a block of data in which the first returned byte indicates data-length. If the BMC attempts to read more bytes than specified by the data-length byte, a bus hang may occur.

Implication**:** If the BMC, operating in I2C mode, reads slave data disregarding the data-length, it will cause the bus to hang**.**

Workaround**:** When BMC acts as Master, it should interpret the first returned data byte received from the device as data-length, and should stop the transaction after reading the specified number of bytes.

Status: No Fix: There are no plans to fix this erratum.

11. SOL Timeout Character Control Byte In EEPROM Image Does Not Function.

Problem**:** SOL (serial over lan) character control can be configured from the EEPROM. Packets from the host to the management console will be sent when either the maximum buffer size or a timeout are reached. However, instead of restarting the timer on every transmit to LAN; the timer is restarted every time a new character arrives from the host. When the transmit rate from the host is slow, the characters will only be sent when the buffer threshold is reached.

Implication**:** Characters transmitted from the host may not arrive at the remote console at the expected refresh rate, but in bursts. This will usually be noticed only at slow rates (for example, manual typing), which is not a use case in SOL.

Workaround**:** None.

Status: No Fix: There are no plans to fix this erratum.

12. Incorrect Number of Retransmissions of Link-Down Alert.

Problem**:** In ASF mode, if Register EEh (Link up delay) is set to 0, then the number of Link Loss packets that are transmitted is one less than the number set in the ASF register EBh.

Implication**:** The number of Link Loss packets that are transmitted is one less than the desired number.

Workaround**:** Set Link up delay to 1, or set retransmission number to be one more than the required retransmission number.

Status: No Fix: There are no plans to fix this erratum.

13. Device Does Not Support PCIe Active State Power Management L1 State (ASPM L1).

Problem: When the device is in ASPM L1, the dynamic clock gating mode is active as a power-saving feature. In this instance, the activating condition for dynamic clock gating is erroneous, resulting in the DMA clock halting when it should be operating.

When both ASPM L1 and ASPM L0 are enabled, and the PCIe interface is set to the x1 mode, the device might cause the PCIe interface to stop responding during the ASPM L1 entry handshake.

Implication: While the device is in ASPM L1 mode, the DMA clock is halted, thus an initiated LSC (Link Status Change) interrupt will be held until the clock is restarted.

While the device is at ASPM L1, and a single packet is received, the packet will be fully DMA'd to the host, but the clock may halt before the write-back is finished, resulting in packet loss.

ASPM L1 must not be enabled.

Workaround: Disable ASPM L1; a device connected to I/O Control Hub 7 (ICH7). Disabling ASPM L1 will prevent the DMI link between ICH7 and the Memory Control Hub (MCH) from entering ASPM L1.

Disable Dynamic Clock gating; this is controlled by EEPROM Word 0xF bit 3. This bit should be always be set to 0 when ASPM is used. EEPROM images based on dev_starter image 5.6 or older have this bit enabled. EEPROM images based on dev_starter image 5.7 or later have this disabled by default.

Advertise ASPM L0s support only; ensure bits 3:2 in Word 0x 1A are set to b01. EEPROM images based on dev_starter image 5.6 or older have these bits set to advertise to the system that ASPM L1 is supported in the device. EEPROM images based on dev_starter image 5.7 or later do not advertise ASPM L1 support. Please note the system decides whether to put the device in ASPM L1

Status: No Fix: There are no plans to fix this erratum.

14. XOFF from Partner Can Prevent Flow-Control (XON/XOFF) Transmission.

Problem: When the device transmitter is paused (by reception of an XOFF packet from the link partner) while data is being processed to be transmitted, both the transmission of normal packets and the outbound XON/XOFF frames (resulting from Receive Packets Buffer level and Flow-Control Thresholds) are paused. Normally, the link partner's XOFF packets pauses the LAN controller for a finite time interval, after which outbound XON/XOFF's (due to the Receives-PacketBuffer being full) can be sent again.

Implication: If the transmitter is paused and the receive FIFO XOFF threshold is reached, the transmission of the XOFF frame does not occur and the Receive FIFO overrun may potentially occur, resulting in lost packets. This is only expected to be seen with an abnormally high pause time from link partners XOFF packet(s).

Workaround: To minimize the likelihood of a Receive FIFO overrun, Receive Flow-Control Thresholds should be based on the expected maximum pause interval in the link partner's XOFF packet. This has been implemented in the Intel drivers.

Status: No Fix: There are no plans to fix this erratum.

15. Missed RX Packets.

Problem**:** When the device operates with multiple-requests or Large Send enabled, there could be receive packet loss.

When the Tx FIFO is full, the Tx flow may block the host DMA interface of the device. When the transmission of packets is prevented for a long time, due to capture effect or very long backoff in half-duplex, the transmit FIFO is filled and the fetch of Rx descriptors is prevented also. This will prevent the release of the packets from the Rx FIFO to the host, causing the Rx buffer to overflow and the loss of incoming packets. This is a temporary state that will be released once the transmit side is be able to empty the Tx packet buffer.

Implication**:** There could be some packet loss in the Rx path if the transmission of packets is prevented for a long time. Normally, if this occurs, these packets will be re-transmitted by upper-layer protocols.

Workaround**:** None

Status: No Fix: There are no plans to fix this erratum.

16. Tx Stops During Host Management Stress in 10 Mbps Half-Duplex.

Problem**:** When the device operates in 10 Mbps Half-Duplex and a packet from management is involved in excessive collisions while both HOST and MNG have a packet ready in the Tx pipe, the Tx process could get into an abnormal state resulting in Tx sticking.

Implication**:** This was found in an abnormal use condition when activating IDE-R together with host stress traffic. In normal operating mode, this condition was not seen.

Workaround**:** The “transmit stuck” state will be released by a software reset.

Status: No Fix: There are no plans to fix this erratum.

17. Device Overwrites Port A LAA to Default Value Due to Port B Software Reset.

Problem: When the LAA (local administrated address) is set by the driver on one port and the second port driver initiates resets, the LAA on the first port also gets resets and the default MAC address is loaded from the EEPROM.

Implication: The driver on the second port is not aware to the fact that its LAA was changed and packets addressed to the LAA will be lost.

Workarounds: When using LAA, the driver should set an additional MAC Address filter (for example, RAR[1]) to the LAA value, so if RAR[0] is overwritten, incoming packets will be accepted by the additional filter. In addition, the driver should poll the value of RAR[0], and, if it detects the RAR[0] value is reset to the EEPROM value, it should reload it with the correct LAA value.

Still, with this workaround the WoL magic packet may not work if a port (for example, port 1) that is enabled for WoL uses a LAA. The problematic scenario is when Port 1 goes to D3 state after checking that it's RAR[0] value is correct. Port 0 goes to D3 state and performs reset for its port, causing the Port 1 LAA to be overwritten again. The port 1 driver is already down, so it does not know this and cannot update this again. As the magic packet WoL uses RAR[0] only, magic packets to port 0 will not wake up the system.

This has been implemented in the Intel drivers.

The following information shows how the workaround can be implemented:

- A boolean flag named *laa_is_present* is added to the adapter structure to identify to the driver that the workaround is to be applied.

```
struct e1000_hw {
  ...
  boolean_t laa_is_present;
```

- When the driver changes the local MAC address, *laa_is_present* is set, if using an 82571, and the new MAC address is written to a redundant slot in the receive address table. In this example, entry 14 is used. Entry 15 should not be used as it may be used by the management functions. In this example, e1000_rar_set() is a shared code function used to update the RAR registers.

```
/* With 82571 controllers, LAA may be overwritten (with the default)
 * due to controller reset from the other port. */
if (adapter->hw.mac_type == e1000_82571) {
    /* activate the work around */
    adapter->hw.laa_is_present = 1;

    /* Hold a copy of the LAA in RAR[14] This is done so that
     * between the time RAR[0] gets clobbered and the time it
     * gets fixed (in e1000_watchdog), the actual LAA is in one
     * of the RARs and no incoming packets directed to this port
     * are dropped. Eventaully the LAA will be in RAR[0] and
     * RAR[14] */
    e1000_rar_set(&adapter->hw, adapter->hw.mac_addr,
                  E1000_RAR_ENTRIES - 1);
}
```

- Periodically, for example in the drivers watchdog function, RAR(0) should be updated with the changed LAA as it may have been rewritten by a reset on Port B.

```
/* With 82571 controllers, LAA may be overwritten due to controller
 * reset from the other port. Set the appropriate LAA in RAR[0] */
if (adapter->hw.mac_type == e1000_82571 && adapter->hw.laa_is_present)
    e1000_rar_set(&adapter->hw, adapter->hw.mac_addr, 0);
```

Intel drivers share some common functions, which have been adapted to this issue:

- e1000_rar_set() is used to update the RAR registers. No changes are required to adapt to this issue, but it is the function used by the following functions.
- e1000_init_rx_addrs() is used to initialize the receive address registers by updating RAR(0) and clearing the remaining RARs. It has been adapted to reserve a spot for the redundant LAA.

```
void
e1000_init_rx_addrs(struct e1000_hw *hw)
{
   uint32_t i;
   uint32_t rar_num;
   /* Setup the receive address. */

   e1000_rar_set(hw, hw->mac_addr, 0);

   rar_num = E1000_RAR_ENTRIES;

   /* Reserve a spot for the Locally Administered Address to work around
    * an 82571 issue in which a reset on one port will reload the MAC on
    * the other port. */
   if ((hw->mac_type == e1000_82571) && (hw->laa_is_present == TRUE))
      rar_num -= 1;
   /* Zero out the other 15 receive addresses. */
   for(i = 1; i < rar_num; i++) {
      E1000_WRITE_REG_ARRAY(hw, RA, (i << 1), 0);
      E1000_WRITE_REG_ARRAY(hw, RA, ((i << 1) + 1), 0);
   }
}
```

- e1000_mc_addr_list_update() is used to initialize the multicast address registers and the receive address registers. It has been adapted to reserve a spot for the redundant LAA.

```
void
e1000_mc_addr_list_update(struct e1000_hw *hw,
                          uint8_t *mc_addr_list,
                          uint32_t mc_addr_count,
                          uint32_t pad,
                          uint32_t rar_used_count)
{
   uint32_t hash_value;
   uint32_t i;
   uint32_t num_rar_entry;
   uint32_t num_mta_entry;

   /* Set the new number of MC addresses that we are being requested to use. */
   hw->num_mc_addrs = mc_addr_count;
```

```
    /* Clear RAR[1-15] */
    num_rar_entry = E1000_RAR_ENTRIES;

    /* Reserve a spot for the Locally Administered Address to work around
     * an 82571 issue in which a reset on one port will reload the MAC on
     * the other port. */
    if ((hw->mac_type == e1000_82571) && (hw->laa_is_present == TRUE))
        num_rar_entry -= 1;

    for(i = rar_used_count; i < num_rar_entry; i++) {
        E1000_WRITE_REG_ARRAY(hw, RA, (i << 1), 0);
        E1000_WRITE_REG_ARRAY(hw, RA, ((i << 1) + 1), 0);
    }

    /* Clear the MTA */
    num_mta_entry = E1000_NUM_MTA_REGISTERS;
    for(i = 0; i < num_mta_entry; i++) {
        E1000_WRITE_REG_ARRAY(hw, MTA, i, 0);
    }
    /* Add the new addresses */
    for(i = 0; i < mc_addr_count; i++) {
        hash_value = e1000_hash_mc_addr(hw,
                                        mc_addr_list +
                                        (i * (ETH_LENGTH_OF_ADDRESS + pad)));

        /* Place this multicast address in the RAR if there is room, *
         * else put it in the MTA
         */
        if (rar_used_count < num_rar_entry) {
            e1000_rar_set(hw,
                          mc_addr_list + (i * (ETH_LENGTH_OF_ADDRESS + pad)),
                          rar_used_count);
            rar_used_count++;
        } else {
            e1000_mta_set(hw, hash_value);
        }
    }
}
```

18. Enabling Or Disabling RSS in the Middle of Received Packets May Stop Receive Flow.

Problem**:** Enabling RSS consists of setting both the Multiple Receive Queues Enable bit in MRQC and the Packet Checksum Disable bit in RXCSUM. Changing these settings while there is data in the receive data FIFO could cause the receive DMA to hang. There may be data present in the receive data FIFO even before the driver initialization is executed if the manageability firmware routes some packets to the host using MANC2H.

Implication**:** No data received.

Workaround**:** The driver should implement the following sequence during initialization if RSS is used:

- Set PBS[31] to disable the receive FIFO.
- Perform a software reset to clear the receive FIFO.
- Set up RSS.
- Write RDFHS = (PBA[5:0] << 7) - 1
- Clear PBS[31].
- Clear RDFHS.
- Set RCTL.EN to enable packet reception

Status: No Fix: There are no plans to fix this erratum.

19. Packets with IPV6 Tunneled in IPV4 and with a Certain Value of Last IP Options Will Have an Incorrect RSS Hash Value.

Problem**:** When IPV6-tunneled-in-IPV4 packets are received, IP option with data is present, and the last byte of IP option is 0x08, an incorrect value of RSS hash (it will be 0x0), queue, and CPU numbers may be calculated.

Implication**:** When working with RSS, the platform uses the RSS hash to do TCP context lookup and has no way of recovering if the RSS hash value is incorrect. In this case, it will drop the packet, and possibly reset the connection.

In addition, this packet may be directed to a wrong queue and wrong CPU.

Workaround**:** If RSS hash value is 0 and PKTTYPE = 3, 4, 9 or 0xA, check IP length. If options are present, do not indicate an RSS hash value to the stack. The TCP stack will calculate the RSS hash value for a TCP packet, which will prevent it from being dropped.

This has been implemented in the Intel drivers.

Status: No Fix: There are no plans to fix this erratum.

20. Formed and Invalid /C/ Code Handling on the SerDes Interface.

Problem**:** The device responds improperly to certain invalid sequences on the SerDes interface, which include comma characters different than k28.5 or symbols with inverted disparity.

Implication**:** The device may:

- Achieve link when it shouldn't
- May not restart auto-negotiation when it should
- In normal operation the comma used is k28.5; inverted disparity should not happen on a normal system.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

21. False Detection of an idle_match Condition on the SerDes Interface.

Problem: The idle_match function is used during the auto-negotiation process for 1000 BASE-X applications (SerDes). This function continuously indicates whether three consecutive /I/ ordered sets have been received and it is observed when moving from IDLE_DETECT state to LINK_OK state within the auto-negotiation state machine.

Though there are not three consistent /I/ symbols (that is,. there is some combination of /I/ and other symbols), the device can incorrectly set the idle match to true.

Implication: This failure should not be seen in normal-use cases where there are many consecutive /I/ symbols in the auto-negotiation process. However, if the erroneous case occurs, the auto-negotiation will continue and lock on the next /I/ pattern.

Workaround: None

Status: No Fix: There are no plans to fix this erratum.

22. Ability Match and Acknowledge Match on the SerDes Interface.

Problem: In the 1000BSE-X state machine, the device does not reset its match count upon reception of /I/ ordered sets in between /C/ ordered sets.

Implication: None: In normal operation, the specific sequences do not occur.

Workaround: None

Status: No Fix: There are no plans to fix this erratum.

23. Frames with Alignment Errors.

Problem**:** The device discards a frame with extra bits. According to IEEE 802.3 2002 section 4.2.4.2.1, a frame containing a non-integer number of octets should be truncated to the nearest octet boundary. After the test frame is truncated, the resulting frame should be accepted as a valid frame.

Implication**:** The device improperly discards frames. This condition occurs rarely in normal operation.

Workaround**:** None

Status: No Fix: There are no plans to fix this erratum.

24. Inter-Frame Spacing (10/100 Half-Duplex Mode Only).

Problem**:** In the 10/100 half-duplex mode (only), the device uses more than 6.4 µs for interFrameSpacingPart1. It does not force collisions according to the IEEE 802.3 standard.

Implication**:** Instead of following 802.3 and initiating transmission independent of carrier sense during interFrameSpacingPart2, carrier sense will still cause a deferral and not cause a forced collision.

Workaround**:** None

Status: No Fix: There are no plans to fix this erratum.

25. Auto-Cross Sample Timer (PHY-related Issue).

Problem**:** The Auto-Crossover State Machine (Auto-MDIX) has two states: MDI_MODE and MDI-X_MODE. The time that should be spent in each mode is defined as an integer multiple of a pseudo-random number and a sample timer, which is defined to be 62 ± 2 ms. The PHY is sometimes waiting for a non-integer multiplication of the 62 ± 2 ms – as defined by the specification.

Implication**:** None

Workaround**:** None

Status: No Fix: There are no plans to fix this erratum.

26. Firmware Reset Occurs when Performing Transactions with a Low Interpacket Gap (IPG) Using Fast Management Link (FML) at 8MHz.

Problem**:** A single firmware reset occurs when FML 8MHz transactions are delivered with an Inter-Packet-Gap (IPG) smaller than 20uSec. Due to speed and memory limitations, buffers of BMC frames being arranged into Ethernet packets are incorrectly released. As a result, a memory error (Null pointer exception) occurs.

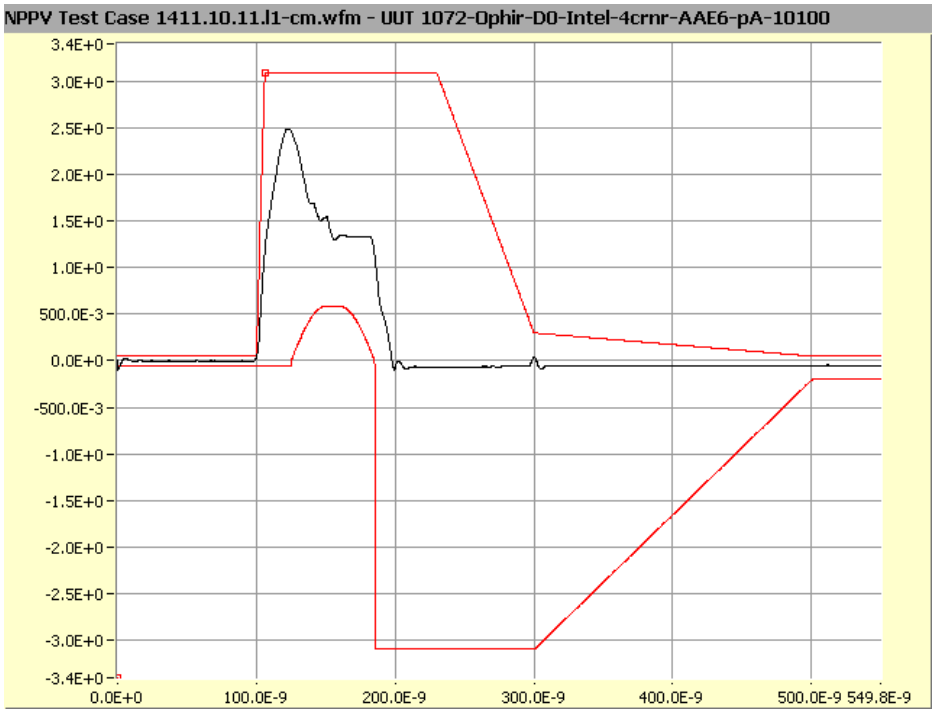
Implication**:** Performance impact. The BMC must retry the corrupted transaction.

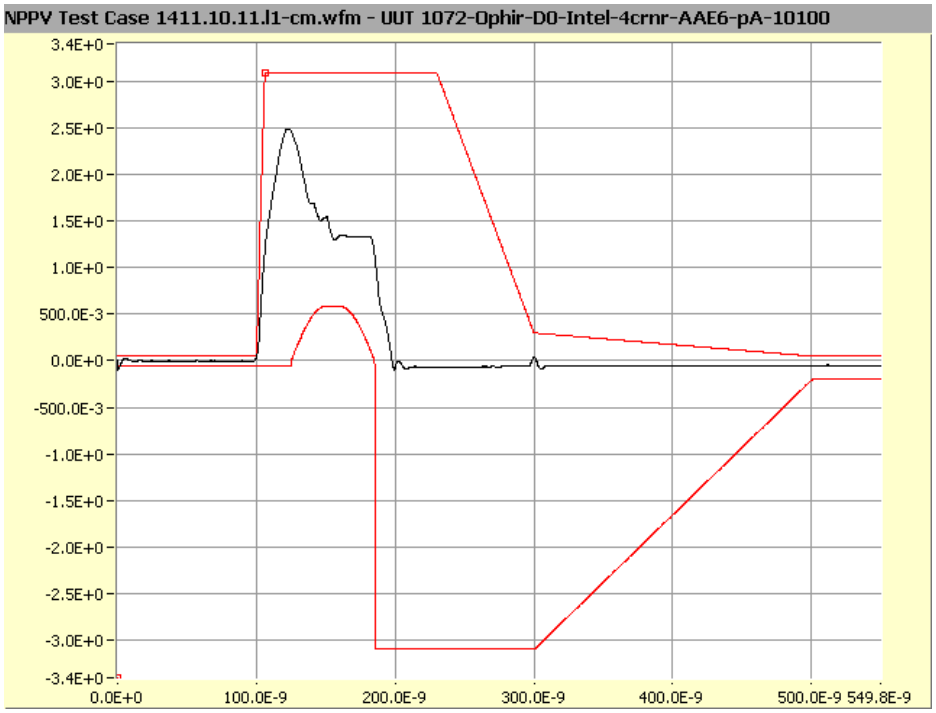
Workaround**:** The BMC should not transmit with an IPG lower than 30usec.

Status: No Fix: There are no plans to fix this erratum.

27. 10base-T Link Pulse Hits the Template Mask Due to Voltage Ripple/Glitch

Problem: The 10base-T link pulse touches the template due to voltage ripple/glitch.





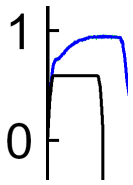
Implication: Compliance with the specification is not complete, however, there is no effect at system level.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

28. 10base-T TP_IDL Template Failure.

Problem: The 10base-T TP_IDL waveform fails the template test with twisted-pair model combined with test load 2.



Implication: There is not full specification compliance. There is no impact on system level performance.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum..

29. BMC Fragments that Are Sent through Two Different SMBus Ports Are Sent Over LAN as a Single Packet.

Problem: When the BMC sends sequential fragments of two packets, using different SMBus ports of the device, they are linked and transmitted as one packet.

Implication: BMC cannot send two non-synchronized packets over two ports.

Workaround: BMC should send only one packet at a time.

Status: No Fix: There are no plans to fix this erratum.

30. Frames with Variations in the Preamble Are Rejected (Copper Only).

Problem: The device (on copper implementations only) rejects frames that contain errors in the preamble.

Implication: The device does not accept the frame with a preamble different than the normal stream (555…55D).

Workaround: None--the rejection of frames with an error in the preamble does not interfere with the reception of valid frames preceding or following the frame containing the error.

Status: No Fix: There are no plans to fix this erratum.

31. Reception of Undersized Frames Affects Good Frame Reception.

Problem: If the device receives a one-byte fragment, then the following first-received frame will be discarded.

Implication: After receiving a frame with a one-byte fragment, the device rejects the following first-received frame.

Workaround: None; this frame will be recovered in a higher-protocol level.

Status: No Fix: There are no plans to fix this erratum.

32. Packet Length-Related Issues.

Problem: The device does not report a length error if an incoming undersized or oversized packet passes the filter criteria.

The device does not truncate the pad from frames with a length field ranging from 0x0000 to 0x002d.

Implication: The device doesn't check the Ethernet length field to verify that the length of the packets matches the value in the length field. Packets with incorrect length field values are not discarded nor reported as required by Section 4.3.2 of IEEE 802.3 2002.

Also, the device does not truncate the pad from frames with a length field ranging from 0x0000 to 0x002d.

Workaround: None; both the data and the pad portion are handled by the higher layers.

Status: No Fix: There are no plans to fix this erratum.

33. When MANC.EN_XSUM_FILTER Is Not Enabled, Received Packets with Wrong UDP Checksum Are Transferred to BMC.

Problem: Received packets with wrong UDP checksum should be silently discarded. UDP checksum filtering could be done by hardware or by firmware. When the hardware filtering option is disabled, that is, MANC.EN_XSUM_FILTER (MACCSR 0x5820 bit 23) is de-asserted, firmware fails to drop the packet and passes it to BMC.

Implication: BMC will receive packets with incorrect UDP checksum.

Workaround: MANC.EN_XSUM_FILTER should be asserted (configurable in EEPROM words 0x4D/0x5D bit 7).

Status: No Fix: There are no plans to fix this erratum.

34. Device Sends Only One XOFF Even if the Link Partner Has Timed Out and It Is Still Congested.

Problem: When Flow Control is enabled, the device should periodically send XOFF packets as long as it is congested to prevent the link partner from sending data and to prevent packet loss. The period of the XOFF packets depends on the XOFF timeout number. The device sends only one XOFF packet per congestion regardless of its congestion status.

Implication: In Flow Control mode, when the device is congested for a time that exceeds the XOFF timeout number, there may be some packet loss. The link partner will wait the XOFF timeout and then continue to send data. In this case, if the device is still congested, the packet will be lost. The reception of the link partner data will cause the device to resend a XOFF packet and the link partner will stop transmission again.

Workaround: Set the maximum timeout number in the XOFF packets to reduce the probability that the device will still be congested after the timeout.

Status: No Fix: There are no plans to fix this erratum.

35. When Wake on LAN (WoL) Is Disabled, the Device Consumes More Than the Specified 20mA.

Problem: When Wake on Lan (Wol) is disabled and external voltage regulators are used (as recommended), the device has been measured consuming up to 100mA. This is a violation of the PCIe and device specifications.

Implication: The specification is for operation with internal regulators, which would allow them to be shut down, thus reducing power consumption. Since the recommended design uses external regulators, the device will operate correctly, but power consumption is greater than specified.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

36. The Device Does Not Correctly Handle Received Nullified Transaction Layer Packets (TLP).

Problem: When a received TLP-end-framing symbol is EDB, and the LCRC is the logical NOT of the calculated CRC (Nullified TLP), it should be "silently" discarded, that is, without setting any error flags. The device discards the TLP correctly, but it also sets a Bad TLP error, and sends a NAK DLLP.

Implication: Nullified TLP is rarely seen in typical operation. If the device receives a nullified TLP, it will send Bad TLP error message and will send a NAK to the nullified TLP. This causes a re-transmission of TLP's that were sent after it (the sequence number is not incremented after a nullified TLP). This should not affect normal operation, since after the re-transmission, traffic will continue normally.

Workaround: None.

Status: No Fix: There are no plans to fix this erratum.

37. Link Down During Receive Flow May Cause Data Corruption.

Problem: When the link fails in the middle of a received packet, the end of the packet may not be set and the next packet, after the link is restored, combines with the previous packet.

Implication: One packet with corrupted data may be received with a good CRC indication.

Workaround: A software reset, after the link is down, will remove the packet that was interrupted by the link failure. This action has been implemented in the Intel drivers.

Status: No Fix: There are no plans to fix this erratum.

38. Incorrect PCIe Configurations Can Be Set by Earlier Versions of dev_starter EEPROM Images (v5.8 and below).

Problem: PCIe configurations can be set incorrectly by EEPROM dev_starter images version 5.8 or older

Implication: The following are the issues that can be seen as result of the PCIe configurations not being loaded correctly:

- PCIe TX Differential Voltage amplitude: Increased to ~1.35V to 1.4V instead of a max of 1.2V. System impact will vary based on the upstream PCIe devices' tolerance to a higher amplitude.
- Absolute Delta of DC Common: During L0 and Electrical-Idle the Delta will increase to approximately 300mV from 100mV. This should not have any functional impact.
- A power-saving feature: When in Electrical-Idle for the PCIe bus, Receive is not enabled.

Workaround: Use an EEPROM image based on dev starter version. 5.9 or above.

Status: Fixed in EEPROM dev starter version. 5.9 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

39. Packets Received with an L2+L3 Header Length Greater than 256 Bytes Can Incorrectly Report a Checksum Error.

Problem: L2/L3 packets with long/multiple next header extensions incorrectly report a Receive checksum error when the length from Destination Address (DA) to the beginning of the TCP/UDP header is greater than 256 bytes.

Implication: A receive checksum error can incorrectly be reported by the device, even if there is no checksum error.

Workaround: When the driver receives a packet with a checksum error reported by the hardware, software should check the L2/L3 header length. If the L2/L3 header length is 256 bytes or greater, software should verify the checksum.

The Intel Windows* and Linux* drivers address this issue by passing packets with bad checksums to the stack for discard or recheck.

Status: No Fix: There are no plans to fix this erratum.

40. PCIe Bus Can Halt upon D3/L1 Entry If There Are Less Than 16 Posted Data (PD) Flow Control Credits (=256byte memory writes).

Problem: The device PCIe transmit bus will halt when entering D3/L1 power states if the upstream device issues less than 16 Posted Data Flow Control credits.

Implication: PCIe bus stops with no communication to the upstream device

Workaround: Upstream PCIe device must issue at least 16 PD type credits.

Status: No Fix: There are no plans to fix this erratum.

41. When APM Enable (WOL) Is Not Set in the EEPROM, It Can Affect the Firmware Load and PCIe Configurations

Problem: If all of the following configuration settings are true:

For the 82571EB:

EEPROM word 0x14/24 (Initialization Control 3), APM Enable (bit 10) are both set to 0b. Dev_starter EEPROM default is set to 1b.

For the 82572EI:

EEPROM word 0x24 (Initialization Control 3), APM Enable (bit 10) is set to 0b. Dev_starter EEPROM default is set to 1b.

For both devices

- PHY/Serdes power down is enabled. EEPROM word 0x0F (Initialization Control 2), PHY Power Down (bit 6) and SerDes Power Down (bit 2) are set to 1b
- Device power down is enabled: EEPROM Word 0x1E, Device Power Down Enable (bit 15) is set to 1b. Dev_starter EEPROM default is set to 1b
- Voltage regulators shut is disabled: EEPROM Word 0xA (Initialization Control Word 1), EE_VR_Power_Down (bit 7) is set to 0b. Dev_starter EEPROM default is set to 0b

Then, if PERST# is still asserted by the system after the EEPROM auto read, which occurs with LAN_PWR_GOOD, configurations that should be loaded from the EEPROM might not be loaded.

Implication: Device might not function properly.

Workaround: When APM is disabled on both ports, de-assert the device Power-Down Enable bit (EEPROM Word 0x1E, bit 15).

Status: No Fix.

42. Traffic on SMBus While Link Is Down Causes Firmware Reset.

Problem: If the Ethernet link is down and traffic is sent to the device via the SMBus, the firmware can be reset and the data is lost.

Implication: The firmware reset will cause the loss of the previous state and disconnect open RMCP sessions.

Workaround: Fixed in EEPROM dev starter image version 5.10 and above. This firmware will discard packets sent while the link is down after timeout. Contact your Intel representative to ensure you have the latest EEPROM release.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

43. SOL Stress Data Integrity Fails with IDER Stress.

Problem: Under heavy SOL stress, along with normal IDER stress, about one in every $3x10^6$ SOL bytes forwarded to the Host is corrupted. No corruption occurs with SoL alone.

Implication: Bytes sent by remote controller may cause unpredictable results in the controlled Host.

Workaround: None.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

44. The 82571EB/82572EI PCIe Transmit Differential Voltage Amplitude Is 1.4V (Maximum of 1.5V) for the First 15ms of Transmission.

Problem: When the PCIe TX starts transmitting after PERST# de-assertion, the first 15 ms will be at approximately 1.4V (maximum of 1.5V). After this, the voltage will be configured to the correct value of approximately 1.1V.

Implication: PCIe TX differential voltage will exceed the specification during this time.

Workaround: None

Status: No Fix.

45. Manageability Software Halts when SMBus Slave Address Is Set to 0x00.

Problem: Attempting to configure only one SMBus slave address in the EEPROM by setting the other address to 0x00 halts manageability software.

Implication: The second slave address cannot be ”disabled.” The BMC will get a notification after events on the second LAN port. If the notification timeout is set to “no-timeout,” the notifications will continue indefinitely and degrade BMC performance.

Workaround: No workaround.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

46. Rx Packet Notification Timeout Does Not Reset After Master Reads Fragment.

Problem: Packets are fragmented to be sent over the SMBus to the BMC. "Notification timeout" is set in EEPROM. If a fragment is not read by the BMC before the timeout expires, the rest of the packet will be dropped. The timeout counter is not reset after every fragment is read, so if the entire packet is not read before timeout expires, the last fragments will be dropped.

Implication: Timeout can be set between 1 ms and 255 ms. Packets can only be read by the BMC if they are completely read within that interval. The expected behavior would be to reset the timeout counter after every BMC read transaction.

Workaround: No workaround.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

47. BMC Configuration Commands are Discarded When There Is Heavy Manageability Traffic Load.

Problem: When there is heavy Rx traffic to the BMC, configuration commands sent to the device are not accepted.

Implication: Denial of Service – the BMC may not be able to change the device's configuration (for example, disable Rx under ARP attack).

Workaround: No workaround.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

48. Duplicate Fragments Might Be Sent to the BMC.

Problem: If the BMC sends two SMBus read transactions with a short delay, the same fragment may be forwarded twice.

Implication: Packets may be forwarded to the BMC in several fragments. Duplicates will cause packets to arrive corrupted on the BMC side.

Workaround: No workaround.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

49. Memory Buffer Leaks Under Heavy SMBus Traffic Load.

Problem: Firmware memory pools, dedicated for SMBus transaction, leak under heavy Tx and Rx traffic until unable to forward ethernet packets.

Implication: Normal manageability traffic, such as KVM and Ping will halt in less than 15 minutes.

Workaround: There is no workaround.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

50. First Two Bytes of a Rx Packet Forwarded to the BMC Might Be Dropped, Degrading Performance.

Problem: Under heavy Rx traffic, the first two bytes of a packet sent over the SMBus can be dropped.

Implication: The BMC will attempt to recover packets by comparing bytes received with the original packet length. This process slows perforance.

Workaround: No workaround.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

51. SMBus Might Hang if the BMC Is Reset in the Middle of a Transaction.

Problem: When the following conditions exist:

- The device is in the middle of an SMBus slave transaction.
- The SMBus master aborts the transaction when the clock is high.
- The device is holding the data low.

  The device doesn't release the data line (because the clock is high) and the SMBus master cannot start a new transaction (data is low) so SMBus hangs.

Implication: When the BMC is reset in the middle of a transaction as described above, it cannot renew the SMBus connection with the devicer or with any other SMBus node sharing the same line.

Workaround: SMBus master should implement some means of releasing the line after reset. For example, toggle the clock at least 9 times so the slave can complete the transaction.

Status: No Fix

52. Certain Malformed IPV6 Extension Headers are not Processed Correctly by the Device.

Problem: Certain malformed IPV6 extension headers are not processed correctly by the device.

Implication: Possible device receive hang if these malformed IPV6 headers are received.

Workaround: Set bit 16 (IPv6_ExdIS) in the RFCTL register to disable the processing of received IPV6 extension headers. Note that with this bit set, HW will no longer offload the receive checksums correctly for incoming frames with IPV6 extension headers, and SW will need to account for this.

Status: No Fix.

53. Completion with CA or UR Status Is Considered Malformed.

Problem: If the device receives a completion with CA (Completer Abort) or UR (Unsupported Request) status, and an all-zero length field, it will recognize the completion as a malformed completion. According to the PCIe specification, this completion is **not** malformed.

Implication: If enabled, an error message will be sent upstream (fatal/non-fatal, as implied by the severity of a malformed TLP error). Default is fatal

Workaround: None.

Status: No Fix

54. HMAC Calculation For RMCP+ Session Establishment Is Incorrect.

Problem: The device does not include the “Name only lookup” bit in the RAKP Open Session request.

Implication: If a RMCP+ utility sets this bit, the resulting HMAC calculation for the utility and the controller will not match and the session establishment process will fail..

Workaround: Set the 'Name only lookup' bit to zero (0).

Status Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

55. SOL Payload Fails to Activate with Encryption Activation Bit Set When Session Was Not Established with Encryption.

Problem: If a RMCP+ session was established by the device, and the Activate Payload command is sent with bit 7 of Byte 3 (Encryption Activation) set to a 1 (one), the controller should ignore it because encryption was not negotiated; instead it fails to activate the payload.

Implication: If using a 3rd party utility such as IPMITool and the device is configured to be the session owner, a SOL session cannot be established

Workaround: Set this bit only if encryption was negotiated.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

56. User Password Not Being Used (Instead of the Kg) when Calculating the SIK.

Problem: While the controller is the session owner, when calculating the SIK, it does not use the user password in place of the Kg, as called for in the IPMI 2.0 specification

Implication: If using a 3rd party utility such as IPMITool and the device is configured to be the session owner, a session cannot be established if a password is used and the Kg is NULL.

Workaround: Use NULL password and NULL Kg, or always configure a Kg.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

57. Firmware Resets While Link Is Down

Problem: Firmware resets when a BMC attempts to send more than one packet while the link is down.

Implication: If the link is lost and more than one packet is attempted to be transmitted, any configuration the BMC performed (such as automatic ARP response MAC and IP Address) will be lost when the reset of the firmware occurs.

Workaround: The BMC can issue the Read Status Command to determine if the link has been lost.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

58. Integrity Value in RMCP+ session establishment

Problem: Incorrect creation/validation of the integrity value in RMCP+ session establishment.

Implication: When the 82571EB/82572EI is the session owner, the RMCP+ session establishment process uses the incorrect key for the calculation and validation of the RAKP integrity check value. The 82571EB/82572EI uses the SIK instead of the K1 key (please refer to the IPMI 2.0 specification for more information on RAKP messages and keys).

The Intel Redirection SDK has the same defect in it; as such, a RMCP+ session can be properly established using the SDK, however other utilities such as IPMITool will fail when a session is established due to the integrity check failure.

Workaround: RMCP+ session establishment can be modified in the user's RMCP+ utility to match the error within the 82571.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

59. Username in RAKP1 Message Must Be Padded to 16 Bytes

Problem: If the 82571EB/82572EI is the RMCP+ session owner, the username in the RAKP1 message must be padded to 16 bytes, or the session establishment will fail.

Implication: Any attempt to establish a RMCP+ session when the 82571EB/82572EI is the session owner will fail if the username is not zero padded to 16 bytes, despite the fact that the username length is part of the message.

Workaround: The RMCP+ software must ensure that the username in the RAKP1 message is zero padded to 16 bytes.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

60. Device Accepts Invalid User Name when RMCP+ Session Owner

Problem: When the 82571EB/82572EI is the session owner, the RMCP+ session establishment process incorrectly accepts an invalid (unconfigured) username as part of the session establishment process if the "Name Only Lookup" bit is not set in RAKP 1 message.

Implication: This is a possible security breach; if this bit is not set the user, is not validated at all.

Workaround: Ensure the RMCP+ session establishment process for the user's application sets the "Name Only Lookup" bit.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

61. Configuring RMCP+ Password from the BMC

Problem: Configuring the RMCP+ password from the BMC (via the SMBus/FML connection) requires a system reset in order to take effect.

Implication: If the BMC configures a password using the "Update User Password" command, the 82571EB/82572EI must be reset in order for the new setting to be used.

Workaround: There is no work-around for this issue.

Status: Fixed in EEPROM dev starter version. 5.12 and above. Contact your Intel representative to ensure you have the latest EEPROM release.

62. "Update User Password" Command Incorrectly Accepts Less Than 20 Bytes of Data

Problem: The "Update User Password" command incorrectly accepts less than 20 bytes of user password data.

Implication: If the "Update User Password" command is used over the SMBus/FML connection and does not zero pad the User Password field of the command to a full 20 bytes, the command will be accepted, however the password stored may be corrupted within the EEPROM image where it is stored, making it impossible to properly establish a SOL/IDER RMCP+ session.

Workaround: The BMC must ensure the password field of the "Update User Password" is zero padded to 20 bytes.

Status: No Fix.

63. Byte Enables 2 and 3 Are Not Set on MSI Writes

Problem: MSI (format code definition Message Signal Interrupts) writes on the 82571EB will not have the upper two Byte Enables (BE) set.

Implication: The PCI specification requires Byte Enables 2 and 3 to be set even though that data will always be zero. Because the 82571/82572 does not set these Byte Enables, MSI writes fail to generate interrupts on systems with chipsets that have been designed to require these Bytes Enables to be set. This errata only applies when MSI is supported and enabled by the system and OS.

Workaround: None, As long as MSI is being used, Byte Enables 2 and 3 will not be set.

Status: No Fix.

64. Wakeup Event Occurs on Magic Packet that Doesn’t Pass Address Filter

Problem: The 82571/82572 receives a magic packet that didn’t pass address filtering. The 82571/82572 will generate a wakeup event at the next packet if the next received packet (non-magic packet) is accepted according to the address filtering scheme.

Implication: The 82571/82572 may wake the system on a non-wakeup packet.

Workaround: None.

Status: No fix.

65. PCIe: SKP ordered set resets Training Sequence (TS) counter.

Problem: If a SKP ordered set is received during a TS1 or TS2 sequence, the TS counter is cleared. This will generally not be a problem since the upstream device should transmit at least 16 TS2 ordered sets, and the 82571/82572 only needs to detect eight consecutive TS2 ordered sets to complete the Recovery process, so a single reset of the counter will not cause a failure. A failure can occur if the upstream device is non-compliant and transmits fewer than 16 TS2 ordered sets. In this case, the 82571/82572 could fail to complete the Recovery process and then the PCIe link would go down.

Implication: There should be no failure when the upstream device functions according to the PCIe spec. If the upstream device is non-compliant, this issue could result in a Surprise Down error.

Workaround: None.

Status: No Fix

66. Internal Copper PHY: Test Equipment May Report Master/Slave Device Doesn't Correctly Implement Master/Slave Resolution.

Problem: When the internal Copper PHY is operating in 1000 Mbps forced slave mode, illegal data may be detected from the device during the transition from 10 Mbps mode (auto negotiation) to 1000 Mbps mode after master/slave resolution is complete.

Implication: Test equipment checking for compliance of Master/Slave resolution may report failures when the device is in Force Slave mode. In Forced Slave mode, the device should not transmit any 1000 Mbps signals, which it is does not. However some test equipment looks for any activity sent from the device in forced slave mode and considers this a failure instead of looking for valid 1000 Mbps signals. Therefore, the illegal data may results in failures reported by test equipment.

Internal validation shows the device compiles with IEEE 802.3 Table 40-5; for all configurations, the device resolves to the correct defined mode

Workaround: None.

Status: No Fix

67. 82571EB-82572EI Improperly Implements the Auto-Negotiation Advertisement Register.

Problem: The 82571EB-82572EI improperly transmits the Link Code Word due to a write to register 4. The Link Code Word improperly switch immediately, which corresponds to a write to register 4. Link Code Word bits behaved as required with the following notes.

Implication: Bits 4.7 and 4.8: Always set in the base page transmission.

Bit 4.9: This bit represents 100BASE-T4 support by the local device. The 82571EB-82572EI does not support T4. It is unlikely that the Auto-Negotiation feature of the 82571EB-82572EI would be used in an implementation to advertise the presence of a separate T4 physical device within the system implementation. Therefore, the fact that this device does not allow T4 to be advertised is insignificant.

Bit 4.15: The 82571EB-82572EI always supports Next Page (regardless the value of bit 4.15). When bit 4.15 is set to "one," the 82571EB-82572EI requires Register 7 (AN Next Page Transmit Register) to be written to complete the Next Page Exchange. In this case however, the 82571EB-82572EI's Next Pages do not correspond to Register 7, but contain valid 1000BASE-T Next Pages.

Workaround: Any write to register 4 should be followed with a restart of Auto-Negotiation by setting bit 0.9.

Status: No Fix

68. PCIe: Reception of Completion That Should Be Dropped May Occasionally Result In Device Hang or Data Corruption

Problem: This erratum can occur when the 82571/82572 PCIe receives a completion that should be dropped, while the 82571/82572 is starting a new request with the same TAG as the completion.

On an error-free PCIe link, this situation should never occur since the 82571/82572 does not assert a second request with the same tag as an outstanding request.

Errors that could cause this failure:

- The TAG of a completion is corrupted due to noise on the line. This completion packet will be dropped due to LCRC error, but it could cause a failure if by chance a new request is asserted with the corrupted TAG value at the same time.
- On some platforms, it has been observed that when the upstream switch port transitions the link to L0s, the 82571/82572 occasionally responds with a NAK as a result of noise on the line. This NAK could cause a completion to be replayed. The 82571/82572 will drop the duplicate packet based on the sequence number. However, the failure could occur if a new request is being asserted with the same TAG as the duplicate completion.
- An edge case of ACK timers results in a replay of a completion. This could cause the same case as above.

Implication: When the failure occurs, the actual completion data from the new request will be corrupted. The implications of this corruption of the read data depend on the type of request the 82571/82572 was starting to send and are described below:

- TX descriptor with TSO – 82571/82572 offload machine may hang.
- TX data or Tx descriptor without offload – 82571/82572 will transmit a packet on the network with invalid data but a valid CRC
- RX descriptor – 82571/82572 will DMA a receive packet to the wrong Memory address.

Workaround: Disabling L0s in the switch port to which the 82571/82572 is connected will prevent the duplicate completions caused by L0s. Keeping bit 13 "ACK/NACK Scheme", word 0x1A " PCIe Initialization Configuration 3" set to 0 in the EEPROM image will minimize the chances of an ACK timeout.

Status: No Fix.

69. Receive packet delayed when using RDTR or RADV register.

Problem: When using the RDTR and/or RADV timer mechanisms, there could be a situation where the write-back timer is incorrectly disabled, which prevents the write-back of a receive descriptor until another packet arrives.

Implication: No packet loss will occur. There may however, be a large delay between the time an Rx packet is received in the device and the time the descriptor is written back to memory, and finally an interrupt generated.

Workaround: It is recommended that the RDTR and RADV registers not be used for moderating Rx interrupts. The preferred solution is to use the Interrupt Throttling Register; ITR.

Status: No Fix

70. 82571/82572 Overwrites Transmit Descriptors in Internal Buffer.

Problem: This erratum occurs when the internal transmit descriptor buffer is nearly full of descriptors. If the free space in this buffer is smaller than the system cacheline, the calculation of the size of the descriptor fetch may be incorrect.

Implication: Corruption of the transmit descriptor ring; can cause a system crash. In most applications, the descriptors will be written back as soon as the data has been read and they will not be accumulating in the internal buffer, therefore this issue will not be seen. However, in an application where system events such as PCIe Flow Control prevent the immediate write-back of descriptors, the descriptor buffer could fill up and this issue could be seen.

Workaround: The driver should keep track of the difference between the Transmit head and tail and make sure the difference between tail and head is never more than the value shown below.

Cacheline	Maximum Value (TDT-TDH)
32 Bytes	62
64 Bytes	60

128 Bytes	56
256 Bytes	48

Status: No Fix

71. Link Indication: LED Remains On In D3 Power State in SerDes Mode.

Problem: The LED might remain on in D3 power state when SerDes power down is enabled (EEPROM word 0xF, bit 2; register CTRL_EXT 0x0018, bit 18). If a link is established when the device enters D3 power state and the LED mode is programmed to reflect LINK indication, the LED remains on even though the SerDes interface powers down.

Implication: LED incorrectly reflects link is up when there is no link (as SerDes is powered off).

Workaround: Set CTRL.LRST (0x0000, bit 3) before putting a function in D3. This brings the link down and turns off the LED; this bit is reloaded from the EEPROM when the device transitions back to D0.

Status: No Fix. There are no plans to fix this erratum.

72. PCIe: Missing Replay Due to Recovery During TLP Transmission

Problem: If the replay timer expires during the transmission of a TLP, and the LTSSM moves from L0 to Recovery during the transmission of the same TLP, the expected replay does not occur. Additionally, if the replay timer is disabled, no further replays will occur unless a NAK is received.

Implication: This situation should not occur during normal operation. If it does occur while the upstream switch is waiting for a replay, the result could be a Surprise Down error.

Workaround: None.

Status: No fix planned.

73. PCIe: LTSSM Moves from L0 to Recovery Only When Receiving TS1/TS2 on All Lanes

Problem: According to the PCIe specification, the LTSSM should move from L0 to Recovery if a TS1 or TS2 ordered set is received on any configured Lane. The Ophir LTSSM only moves from L0 to Recovery if a TS1 or TS2 ordered set is received on all configured lanes.

Implication: This situation should not occur during normal operation since the upstream switch will transmit the TS1 or TS2 ordered sets on all lanes at the same time. If it does occur due to a broken lane, the result could be a Surprise Down error.

Workaround: None.

Status: No planned fix

74. Missing Interrupt Following ICR Read

Problem: If the Interrupt Cause Register (ICR) is read when at least one bit is set in the interrupt mask register and INT_ASSERTED is 0, a new interrupt event occurring on the same clock cycle as the ICR read is ignored.

Implication: Missed interrupts leading to delays in responding to interrupt events. Specifically, this can cause a delay in processing a received packet.

Typically, the ICR is only read in response to an interrupt so this problem would not occur. However, when using legacy interrupts and sharing interrupts between devices, the software may poll all the devices to find the source of the interrupt, including those devices that did not assert an interrupt. There may also be other situations in non-Intel drivers where ICR is polled even when no interrupt has been asserted.

Workaround: If reading ICR when there is no active interrupt cannot be avoided, clear the mask register (by writing 0xffffffff to IMC) before reading ICR. Note tHat in this case the ICR will be cleared when read even if INT_ASSERTED is 0.

Status: No planned fix

75. Tx Packet Lost After PHY Speed Change Using Auto-negotiation

Problem: If the PHY establishes a link at 10/100 Mb/s and then auto-negotiation is re-started and a link is established at 1 Gb/s without resetting the PHY in between, the first one-to-three Tx packets provided by the MAC might not be transmitted.

Implication: This situation is generally seen during testing where the speed of the link partner is intentionally changed.

During normal operation, the packet loss could occur if the cable was moved to a different port. In most cases, the higher layers would handle the packet loss and it would not be visible to the end user.

Workaround: If it is critical that no packets be lost, the software driver can be modified to perform a PHY reset each time it is notified of a speed change.

Status: No planned fix

76. Tx Data Corruption When Using TCP Segmentation Offload

Problem: When using TSO, a situation can occur where a PCIe MRd request is repeated with the same address, resulting in data corruption. At the end of the TCP packet, the Tx DMA hangs because the length doesn't match. This can only occur when the following are true:

- The first buffer of the packet is larger than [3 * (max_read_request - 4)].
- There is a 4 KB boundary within 64 bytes following the end of the header bytes in the buffer.

Implication: Possible data corruption since a TCP packet is transmitted containing the wrong data but with the correct checksum.

Data transmission halts as the Tx DMA module enters a hang state.

Workaround: The failure can be avoided by ensuring at least one of the following:

- The buffer containing the headers should not be larger than [3 * (max_read_request - 4)]. To meet this requirement even for the minimum value of 128 bytes for max_read_request, the buffer should not be larger than 372 bytes.
- The alignment of the buffer containing the headers should be such that there is no 4 KB boundary within 64 bytes following the end of the header bytes. Assuming

standard Ethernet/IP/TCP headers of 54 bytes, this means that the buffer should not start 54-118 bytes before a 4 KB boundary. For example, 128-byte alignment for this buffer could be used to fulfill this condition.

This problem has not been reported when using an Intel Linux* or Windows* driver. Current analysis shows it is very unlikely for a situation to exist that would cause the 82571/82572 to be at risk for the errata when using the Intel Linux or Windows drivers.

Status: No planned fix

77. PCIe: Extended PCIe "Hot Reset" Can Lead to a Firmware Hang

Problem: A PCIe hot reset prevents the firmware from accessing internal registers, including the registers used to access the EEPROM. When an extended PCIe Hot Reset (one second or longer) occurs while the firmware is attempting to initialize itself by reading the EEPROM, the firmware hangs.

This failure occurs only when using NoMNG EEPROM images if a Hot Reset occurs within about two ms after an event that triggers execution of the firmware. Specifically, this failure has been observed when an extended Hot Reset occurs as soon as the PCIe link is established following a PCIe link down event.

Implication: PHY initialization from the EEPROM is performed by firmware, so if the firmware hangs, the initialization is not performed and the Ethernet link might not operate correctly.

Additionally, the software driver might fail to load since the CFG_DONE is not set.

Workaround: One or both of the following:

—Hot Resets should have a duration of less than 950 ms to prevent a firmware hang. It is preferable that Hot Resets be as short as possible to minimize interference with the firmware execution.

—Add a delay of at least several ms between establishing a PCIe link and starting Hot Reset.

Status: No planned fix

78. SerDes: RXCW.RxConfigInvalid Set Incorrectly

Problem: When the device has been receiving a continuous stream of /C/ ordered sets for an extended period of time, the RXCW.RxConfigInvalid may be set as the result of an internal FIFO overflow even if all the input symbols are valid.

Implication: False indication of invalid symbols may cause the driver to disable the link when there is really no problem.

Workaround: Software that uses the RxConfigInvalid bit should account for this behavior. For example, when the RxConfig bit is consistently 1b, it would be reasonable to ignore the RxConfigInvalid bit.

Intel drivers address this erratum for the device by looking to see if the 82571-82572 has Sync and Invalid bit set then read RXCW several times, if Sync and Config both are consistently 1 then ignore Invalid bit and restart Autoneg. This is done when link is down and driver is trying to determine if the link support Auto Negotiation by looking for /C/ ordered set and if /C/ ordered sets are seen then Auto Negotiation is enabled(TXCW.ANE) to try an link up via Auto Negotiation.

Status: No planned fix

79. PCIe: Spurious SDP/STP Causes Packets to be Dropped

Problem: When a spurious SDP or STP symbol is received without a corresponding END symbol, the alignment of the received data presented to the link layer might be incorrect. In this case, any following DLLPs or TLPs are dropped. This situation continues until there is an END symbol received that is not immediately followed by an SDP or STP symbol.

This issue only occurs when the PCIe link width is x1 or x2.

Implication: Usually, this issue causes nothing more than a replay of a few TLPs. The 82571-82572 recovers from this situation autonomously.

If the 82571-82572 is connected to an ICH7, a spurious SDP or STP symbol that occurs just before entering L1 could cause a hang of the PCIe link since the ICH7 does not insert SOS when transmitting PM_Request_ACK DLLPs, so the 82571-82572 does not receive them and never enters L1.

Workaround: If the 82571-82572 is connected to an ICH7, ASPM L1 should be disabled.

Otherwise, no workaround is required.

Status: No planned fix.

8. Specification Clarifications

1. Disable Auto MDI-X for Forced 100BASE-TX Operation.

Clarification: Link may fail if Auto MDI-X is enabled during forced 100BASE-TX mode operation. Since the device does not disable this function automatically, the driver must perform this step. Auto MDI-X can be disabled by clearing PHYREG18.12. Intel's software drivers have been implemented in this way.

Document: *PCIe* Family of Gigabit Ethernet Controllers Software Developer's Manual*.

2. Request Will Not Be Treated as Completion Abort (CA) when the Programming Model Bytes Enable is Violated.

Clarification: The PCI Express specification allows a device to not accept certain requests. This is under the "programming model" cases. The device needs to issue a Completion Abort error if the specific request violates the programming model. As part of its programming model, the device does not support writes with byte enables to specific memory addresses. These writes will be fully executed and will not be treated as Completion Abort.

"CSR writes with partial bytes enables" will be executed in specific address ranges. This scenario will not occur when using the device driver. This functionality is also not needed for the normal operation of the design.

This can be avoided by having no partial bytes enable writing to the device.

3. System-Level EMI Test Can Be Affected by 490MHz Harmonic Seen In 10Base-T Waveform Spectrum.

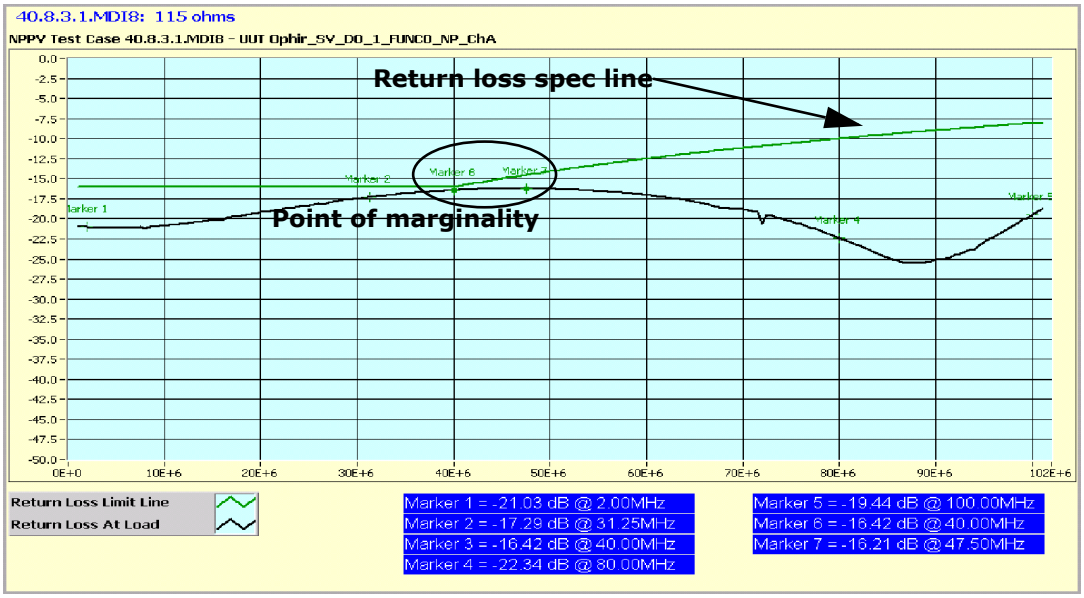
Clarification: This is not a violation of any IEEE specification. This harmonic may, however, contribute to EMI (electro-magnetic interference) in a system-level check at this frequency.

There is no impact on system-level performance.

4. MDI Return Loss Is Marginal Near 40MHz at 115ohm Load.

Clarification: Return loss for 115 ohm load is marginal (Gigabit IEEE specification Section 40.8.3.1) near the frequency of 40MHz. Return loss is acceptable throughout the rest of the frequency spectrum and conforms to the 10Base-T and 100Base-TX specification limits.

IEEE conformance is marginal. There is no impact on system-level performance.



5. PCIe Output Driver Amplitude Can Be Set Incorrectly by the EEPROM.

Clarification: Older versions of theEEPROM images (based on dev_starter image version 5.6 or older) that support Manageability modes (ASF, Pass through, and Super Pass through) can set the PCIe amplitude to an incorrect value.

The latest versions of EEPROM images (based on dev_starter version 5.10 or newer) are required to properly set the PCIe output driver amplitude.

If EEPROMs with the Manageability modes enabled have been used, please contact you Intel representative to ensure you have the latest EEPROM image required for your system.

If you need an updated EEPROM image, it can be obtained from your Intel representative.

6. Only One Port Can Be Disabled at a Time; LAN Disable (LAN0_DIS_N & LAN1_DIS_N)—82571 Only

Clarification: These pins cannot both be low at the same time since this would disable both LAN ports, which is not a valid operating mode.

7. Manageability Modes Not Available When System Is in S5 State when "device power down" Is Activated and APM Is Disabled.

Clarification: If the following configuration is set, the PHY is powered down which prevent a Ethernet link from being established in S5. Therefore the manageability mode is not available when the system is in the S5 state because there is no Ether net link.

For the 82571EB: EEPROM word 0x14/24 (Initialization Control 3) and APM Enable (bit 10) are both set to '0'. Dev_starter EEPROM default is set to '1'

For the 82572EI: EEPROM word 0x24 (Initialization Control 3) and APM Enable (bit 10) are both set to '0'. Dev_starter EEPROM default is set to '1'

For the 82571EB and 82572EI: The following EEPROM settings are left at their default values:

- PHY/Serdes power down is enabled.
- EEPROM word 0x0F (Initialization Control 2), PHY power-down (bit 6) and SerDes power-down (bit 2) are set to 1.
- Device Power down is enabled:
- EEPROM Word 0x1E (Device Revision ID), Device power down disable (bit 15) is set to '1'. Dev_starter EEPROM default is set to '1'
- Voltage Regulators shut is disabled:
- EEPROM Word 0xA (Initialization Control Word 1), EE_VR_Power_Down (bit 7) is set to '0'. Dev_starter EEPROM default is set to '0'
- In order to have the manageability available in S5, "device power down" in EEPROM Word 0x1E (Device Revision ID) (bit 15) must be disabled by setting this bit to '0'. EEPROMs based on dev_starter image 5.10 have this setting disabled by default. Earlier versions had this bit enabled.

Note: All dev_starter EEPROM images have APM enabled.

8. Manageability not supported on SMBus 1.

Clarification: Manageability on SMBus 1 is not supported. This will be reflected in the next release of *82571/82572/631xESB/632xxESB System Manageability Guide*.

9. Support for WOL Concurrently on Both Ports

Clarification: If WOL is enabled on both ports of the 82571EB, then it is possible, in the D3 state, for the device to draw more than the *PCI Bus Power Management Interface Specification Revision 1.1* value of 375 mA on the 3.3Aux voltage rail. In order to always meet this specification, only one port should be enabled for WOL in the 82571EB. Intel Drivers limit the use of WOL to Port 0(Port A) of the 82571EB.

10. LED Modes Based On LINK Speed Only Work in Copper(Internal PHY) Mode

Clarification: LED modes (as defined in Table 5-26 in PCIe* GbE Controllers Open Source Software Developer's Manual) based on LINK speed work only in copper mode, not SerDes mode. This includes the modes LINK_10/1000, LINK100/1000, LINK_10, LINK_100, LINK_1000 and COLLISION.

Designs using SerDes modes requiring a Link up indication should use LINK_UP or LINK/ACTIVITY not LINK_1000. Using these modes results in no issues in using the LEDs to properly indicate the link is up.

11. THERM_Dp (D4) and THEMR_Dn (D5) are reserved and should not be used.

Clarification: In section 3.11 and Table 32 in the Intel® 82571 & 82572 Gigabit Ethernet Controller Datasheet, are references to signals THERM_Dp (D4) and THEMR_Dn (D5), these pins are RESERVED and should not be used in any design. These pins should be left unconnected (floating).

12. TCP Segmentation Offload Operations With Both Transmit Queues Enabled.

Clarification: When using TCP Segmentation Offload (TSO) with both Transmit Queues enabled, bits 6:0 "COUNT" in the TARC0 (0x03840) and TARC1 (0x3940) register must be set to 1 for proper operation. Failure to set COUNT =1 can result in the Transmit flow of the 82571/82572 halting unexpectedly.

13. When Port 0 and Port 1 Are Connected Back-to-Back, the PHY Should Be Reset As Part of the Driver Initialization To Avoid Link Failures.

Clarification: If the PHY is not reset, then both ports might start the Auto-MDI-X protocol behavior at the same exact time. Therefore, both ports will get the same Pseudo Random time out of a power-on reset. As a result, each port powers up with the same configuration of MDI/MDIX. If they are both switching at the same exact time, link does not occur since RX activity is never detected on the receiver (the device turns off the RX circuit on the TX path so as not to falsely establish link from its own link pulses).

On reset, the PHY or when physically connecting the PHY to another device, the psuedo random time of the port is reset and is different from the other port, thus enabling the link to be established. Other factors, such as cable length and silicon variations can have an effect on how close the timings are, but the only time it is an issue is when the connections are back-to-back on the same 82571.

14. PCIe: Completion Timeout Mechanism Compliance

Clarification: If the latency for PCIe completions in a system is above 21 ms and the PCIe completion timeout mechanism is enabled, there can be unpredictable system behavior.

The 82571EB/82572EI complies with the PCIe 1.0a specification for the completion timeout mechanism. The PCIe 1.0a specification provides a timeout range between 50 µs

to 50 ms with a strong recommendation that it be at least 10 ms. The 82571EB/82572EI uses a range of 21-42 ms.

The completion timeout value in a system must be above the expected maximum latency for completions in the system in which the 82571EB/82572EI is installed. This will ensure that the 82571EB/82572EI receives the completions for the requests it sends out, avoiding a completion timeout scenario. If the latency for completions is above 21 ms, this can result in the device timing out prior to a completion returning. In the event of a completion timeout, per direction in the PCIe specification, the device assumes the original completion is lost, and resends the original request. In this condition, if the completion for the original request arrives at the 82571EB/82572EI devices, this will result in two completions arriving for the same request, which may cause unpredictable system behavior.

Therefore, if the PCIe completion latency for a system cannot be guaranteed to be lower than 21 ms, the PCIe completion timeout mechanism should be disabled by setting the GCR.Disable_timeout_mechanism.

For more details on Completion Timeout operation in the 82571EB/82572EI refer to the *Intel®82571EB/82572EI Controller Datasheet* and the *PCIe* GbE Controllers Open Source Software Developer's Manual.*

15. Critical Session (Keep PHY Link Up) Mode Does Not Block All PHY Resets Caused by PCIe Resets

Clarification: D3->D0 transition will cause a PHY reset even in Keep PHY Link Up mode. When Critical Session Mode (Keep PHY Link Up) is enabled (via the SMBUS Management Control command), PCIe resets should not cause a PHY reset. However, the following event will still cause a PHY reset:

Transition from D3 to D0 without a general PCIe reset, i.e. PMCSR[1:0] is changed from 11 to 00 by a configuration write.

Loss of link can cause a loss of the MNG session. These events do not normally occur during a reboot cycle, so it is expected that no effect will be seen in most circumstances.

16. Receiver Detection Circuit Design and Established Link Width.

Clarification: The 82571 receiver detection circuit was designed according to the PCIe Specification Rev. 1.1, which requires that an un-terminated receiver have an input impedance of at least 200 Kohm. PCIe Specification Rev. 2.0 allows the input impedance to be as low as 1 Kohm at input voltages in the range -150 - 0 mV and does not specify a minimum input impedance below -150 mV. As a result, a powered-down receiver lane with low input impedance at negative voltages could be compliant to Rev 2.0 and yet be falsely detected by the 82571 as a terminated lane.

This is normally not an issue since any connnected lanes should be properly terminated within 5 ms after fundamental reset according to the PCIe Specification. However, there are some chipset devices that require significantly more time to prepare the termination and expect the link partner to remain in the LTSSM Detect state as long as none of the lanes are terminated. When used with such devices, the 82571 might falsely detect a receiver on one or more lanes and leave the Detect state. This can lead to not establishing a link or establishing a link that is less than full width.

In this case, it is recommended that:

1. If some of the PCIe lanes are not connected, the Lane_Width field in the PCIe Init Configuration 3 EEPROM word should be programmed to match the actual width of the connection.

and

2. A Hot Reset should be performed after a link has been established in order to force the 82571 to detect the receivers again when they are properly terminated. As a result, a full-width link can be established.

17. Use of Wake on LAN Together with Manageability

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 wakes 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 do not wake the system. However, with NoTCO=1b, broadcast packets, including broadcast magic packets, do not wake the system since they pass the manageability filters and are therefore excluded.

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

The Intel Windows drivers set NoTCO by default.

9. Software Clarifications

1. While In TCP Segmentation Offload, Each Buffer is Limited to 64 KB

The 82571-82572 supports 256 KB TCP packets; however, each buffer is limited to 64 KB since the data length field in the transmit descriptor is only 16 bits. This restriction increases driver implementation complexity if the operating system passes down a scatter/gather element greater than 64KB in length. This can be avoided by limiting the offload size to 64 KB.

Investigation has concluded that the increase in data transfer size does not provide any noticeable improvements in LAN performance. As a result, Intel network software drivers limit the data transfer size in all drivers to 64 KB.

Please note that Linux operating systems only support 64 KB data transfers.

For further details about how Intel network software drivers address this issue, refer to Technical Advisory TA-191.

2. Serial Interfaces Programmed By Bit Banging

When bit-banging on a serial interface (such as SPI, I^2C, or MDIO), it is often necessary to perform consecutive register writes with a minimum delay between them. However, simply inserting a software delay between the writes can be unreliable due to hardware delays on the CPU and PCIe interfaces. The delay at the final hardware interace might be less than intended if the first write is delayed by hardware more than the second write. To prevent such problems, a register read should be inserted between the first register write and the software delay, i.e. “write”, “read”, “software delay”, “write”.