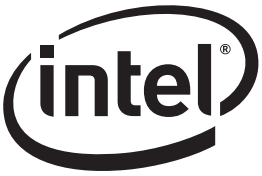


Intel® 82575 Gigabit Ethernet Controller Specification Update and Sighting Information

Networking Silicon

Revision 2.0
October 2008

Revision History

Revision	Revision Date	Description
0.25	Aug 2006	Initial release.
0.30	Oct 2006	Added errata 4 through 14
0.40	Jan 2007	Added errata 15 and 16
1.00*	August 2007	Removed errata fixed in current stepping, previously numbered 4-12 and renumbered the remainder. Added new errata 8 through 16.
1.1	October 2007	Added “SerDes” to application types Corrected Fiber/SerDes Device ID to 0x10A9.
2.0	October 2008	Updated Erratum 13; added errata 17 and 18.

* This revision is numbered 1.00 as part of the device’s release requirements; there are no other revisions between 0.40 and 1.00.

This page is intentionally left blank.

Contents

This page intentionally left blank.

1.0 Preface

This document applies to the Intel® 82575 Gigabit Ethernet Controller. In this document it is commonly referred to as "the device."

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

- *82575 Gigabit Ethernet Controller Product Preview Datasheet*, Intel Corporation.
- *82575 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.

This page left intentionally blank.

2.0 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.
Sightings	Observed issues that are believed to be errata, but have not been completely confirmed or root caused. The intention of documenting sightings is to proactively inform users of behaviors or issues that have been observed. Sightings may evolve to errata or may be removed as non-issues after investigation completes.
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.

Table 2 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.
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.

Name	Description
DS	Data Sheet
DG	Design Guide
SDM	Software Developer’s Manual
EDS	External Data Specification
AP	Application Note

3.0 Product Code & Device Identification

Product Code: :JL82575EB (unleaded), HL82575EB (leaded)

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

Table 3 Markings

Device	Stepping	Top Marking	Q-Specification	Notes
82575EB	A1	JL82575EB	Q899	Engineering Samples
82575EB	A2	JL82575EB	Q900	Engineering Samples (lead free)
82575EB	A2	HL82575EB	Q901	Engineering Samples (lead)
82575EB	A2	JL82575EB	N/A	Production (lead free)
82575EB	A2	HL82575EB	N/A	Production (lead)

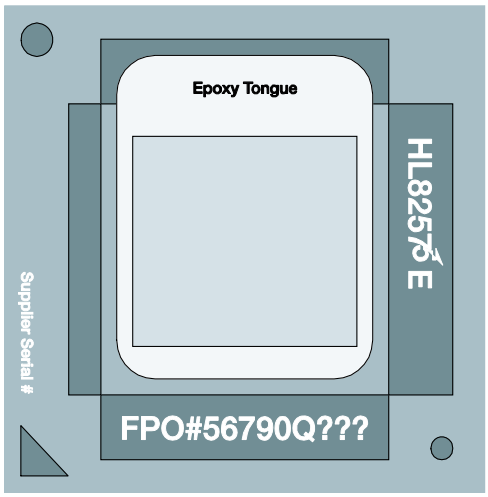
Note: The devices can have a “GB” marking; these devices are used only on Intel network interface adapters. The ”GB” is functionally equivalent to the “EB” version..

Table 4 Revision ID

Device	Vendor ID	Device ID	Revision ID*
82575EB (Copper Applications)	8086	0x10A7	0x0
82575EB (Fiber/SerDes) Applications)	8086	0x10A9	0x0

Table 5 MM Numbers

Product	Tray MM#	Tape and Reel MM#
JL82575EB (lead free)	890991	890990
HL82575EB (lead)	891622	891621



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

Figure 1 Example 82575 Identifying Marks (Similar)

Note: The devices can also have a "GB" marking; these devices are used only on Intel network interface adapters. The "GB" is functionally equivalent to the "EB" device.

4.0 Summary of Table Changes

No.	A0	A1/A2	Plans	Specification Changes	Page
				None	

No.	A0	A1/A2	Plans	Errata	Page
1	X	X	NoFix	Poisoned TLP is reported in all functions instead of target function.	9
2	X	X	NoFix	10BASE-T IDL Template Failure.	9
3	X	X	NoFix	10base-T Link Pulse Hits the Template Mask Due to Voltage Ripple/Glitch.	9
4	X	X	NoFix	Power Management Is Version 2; Should Be Version 3.	10
5	X	X	NoFix	L0s Exit Latency in Link Cap Register Is Not Updated for Common Clock Configuration.	10
6	X	X	NoFix	Wakeup Event Occurs on Magic Packet that Doesn't Pass Address Filter.	10
7	X	X	NoFix	PCIe Differential Return Loss is More than Specified Value.	11
8	X	X	NoFix	SKP (SKIP ordered set) Will Reset TS (Training Sequence) Count.	11
9		X	NoFix	Preamble Error Reception and Recovery.	11
10		X	NoFix	82575 Does Not Correctly Implement Master/Slave Resolution.	12
11		X	NoFix	82575 Improperly Implements the Auto-Negotiation Advertisement Register.	12
12		X	NoFix	In SGMII mode, Counters Incorrectly Increment on Collision.	13
13		X	NoFix	Link LED Remains On after System Power Down and Cable Removed.	14
14		X	NoFix	82575 Improperly Implements Sample Timer.	14
15		X	NoFix	82575 Uses INTB Line When in Single-Function Mode.	14

16	X	X	NoFix	In a Small Number of Cases, the 82575 Sends “Abilities” Instead of “Break Link.”	15
17	X	X	NoFix	Received Packet(s) After Initialization Can Be Corrupted	15
18	X	X	NoFix	PCIe: Reception of Completion That Should Be Dropped, Can Occasionally Cause Device Hang or Data Corruption	16
No.	**A0**	**A1/A2**	**Plans**	**Sightings**	
				None.	
No.	**A0**	**A1/A2**	**Plans**	**Specification Clarifications**	
1	X	X	Doc	PCIe partial Memory Write requests are actually writing the full DW..	16

5.0 Specification Changes

None

6.0 Errata

1. Poisoned TLP is reported in all functions instead of only the target function.

Problem: A fatal message is incorrectly sent in response to a poisoned Transaction Layer Packet (TLP).

Implication: 82575 treats all poisoned memory requests as non-function specific. Instead of reporting in the target function, a fatal error is reported in all functions;. The correct action is to report poisoned requests per function.

Workaround: None.

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

2. 10BASE-T IDL Template Failure.

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

Implication: There is no impact on system level performance.

Workaround: None.

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

3. 10base-T Link Pulse Hits the Template Mask Due to Voltage Ripple/Glitch.

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

Implication: There is no effect at system level.

Workaround: None.

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

4. Power Management Is Version 2; Should Be Version 3.

Problem: Power management version in PMC register is “2” instead of “3.”

Implication: No impact at functional system level.

Workaround: None.

Status: No Fix.

5. L0s Exit Latency in Link Cap Register Is Not Updated for Common Clock Configuration.

Problem: The L0s Exit Latency from theLink Capabilities register (offset 0xAC) will be the same whether the Link Control reg. Common Clock Configuration bit is set or not.

Implication: This function will remain with the default, which is the non-common mode.

Workaround: None.

Status: No Fix.

6. Wakeup Event Occurs on Magic Packet that Doesn't Pass Address Filter.

Problem: The 82575 receives a magic packet that didn't pass address filtering. The 82575 will generate a wakeup event at the next packet, if the next received packet (non-magic packet) is accepted according to address filtering scheme.

Implication: The 82575 may wake the system on a non-wakeup packet.

Workaround: None.

Status: No Fix.

7. PCIe Differential Return Loss is More than Specified Value.

Problem: The PCIe transmitter's differential return loss is -9.7 dB instead of the -10 dB requirement.

Implication: The out-of-specification return loss adds noise to the Tx line. Performance is not affected.

Workaround: None.

Status: No Fix.

8. SKP (SKIP ordered set) Will Reset TS (Training Sequence) Count.

Problem: Upon reception of a SKP, the TS counter will be reset. In link training and recovery, even if the upstream PCIe device sends the correct number of TS, the device may not count all of them if a SKP is sent in between the TS. The 82575 needs 8 consecutive TS1 and then 8 consecutive TS2 to move forward in the LTSSM state machine. If a SKP is sent in between those 8, the internal count of these will start over at 0.

Implication: In Link training and recovery this could result in the upstream device or the 82575 timing out and going back to detect and starting the training process again. If both devices follow the timeouts given the in the PCIe specification, they should self-recover and eventually link.

Workaround: None. Devices should self-recover.

Status: No Fix

9. Preamble Error Reception and Recovery.

Problem: Requirement is for the 82575 to be tolerant of the content of the preamble, that is, any content should be accepted as long is it is not a SFD (Start of Frame Delimiter). The 82575 improperly discards frames with variations in the preamble pattern.

Implication: This problem will only be seen if the link partner is sending a frame with incorrect preambles.

Workaround: None.

Status: No Fix

10. 82575 Does Not Correctly Implement Master/Slave Resolution.

Problem: For some configurations, the 82575 resolves to Master instead of Slave (even though it is in forced-Slave mode). This condition can be detected during the initiation of training by the 82575. This indicates that the 82575 did not remain silent (defined in Slave mode) and that the 82575 resolved to Master mode.

Implication: The equipment that is used in these tests will identify activity as 1 Gb activity if it senses any activity on all four channels. Also, it does not identify 1 Gb idles as 1 Gb training signals. Monitoring the line showed that no training was initiated by the 82575 when the it was resolved to Slave mode.

The activity detected is illegal data that the 82575 transmits during the transition from 10Mbps mode (Auto-Negotioation) to 1000Mbps mode (after Master/Slave resolution is completed). Internal indications (the PHY registers) show that the 82575 complies with IEEE 802.3, Table 40-5; for any configuration, the 82575 resolved to the correct defined mode.

Workaround: None.

Status: No Fix

11. 82575 Improperly Implements the Auto-Negotiation Advertisement Register.

Problem: The 82575 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 82575 does not support T4. It is unlikely that the Auto-Negotiation feature of the 82575 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 82575 always supports Next Page (regardless the value of bit 4.15). When bit 4.15 is set to "one," the 82575 requires Register 7 (AN Next Page Transmit Register) to be written to complete the Next Page Exchange. In this case however, the 82575'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

12. In SGMII mode, Counters Incorrectly Increment on Collision.

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

Name	Definition	Location
RLEC	Lenght error counter	0X4040
CRCERRS	CRC error counter	0x4000
RFC	receive frame counter	0x40A8

Implication: Error counters may not be accurate.

Workaround: None.

Status: No Fix

13. Link LED Remains On after System Power Down and Cable Removed.

Problem: The link-up LED will be set during link-down if a software reset (CSR 0X0000 bit number 26) occurs. The LED will remain on even though power has been removed. This occurs only when the Software Power Down (SPD) is enabled in the EEPROM. The *Smart Power Down Enable* bit is word 0xF, bit 1, which is set by default to 1, which enables SPD.

Implication: The LED will show link-up even when link is down.

Workaround: Disable SPD.

Status: No Fix

14. 82575 Improperly Implements Sample Timer.

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 a multiple of a pseudo-random number and A sample timer, which is defined to be 62 ms ± 2 ms. This violation occurs in ~10% of the runs.

Implication: The time that the PHY is in MDI mode will have a slight deviation from the specified definition.

Workaround: None.

Status: No Fix

15. 82575 Uses INTB Line When in Single-Function Mode.

Problem: Problem occurs when the 82575 is configured with lan 0 to use INTA and lan 1 to use INTB. When lan0 is disabled, there is a switch between the functions and func0 works with lan 1 (single function mode). When interrupt pin register address (0x3D) is read, it shows that the function use INTB, which violates the PCI-e standard of one interrupt line for a single-function device or connector. For a single-function device only INTA# can be used (the other three interrupt lines, used in multi-function mode, have no meaning).

Implication: Interrupt is tagged with INTB instead of INTA.

Workaround: When working with LAN0 disable, change EEPROM Word 0x14, bit 12 and bit 11 to “0”.

Status: No Fix

16. In a Small Number of Cases, the 82575 Sends “Abilities” Instead of “Break Link.”

Problem: When the 82575 is receiving continuous ill-formed /C/ order sets, it should transmit “break link” and not “abilities.” During testing, about 14% of the time, the 82575 sent “abilities” following the link timer expiration.

Implication: Minor specification violation—no system-level performance impact.

Workaround: None.

Status: No Fix

17. Received Packet(s) After Initialization Can Be Corrupted.

Problem: Due to a bug in the receive packet buffer control logic, a corrupted packet or partial packet may be written to the packet buffer before RCTL.RXEN is set. Once RXEN has been set, the first packet provided to the host may have an invalid descriptor and/or corrupted data. In some cases, the packet buffer control is also corrupted, and invalid packets will continue to be transferred to the host indefinitely.

This problem occurs when either of the following conditions is true:

- Manageability is enabled
- Wake on LAN is enabled

Implication: When a single corrupted packet is received, higher layers of the protocol stack will usually discard it, and it will not affect the application. However, if the packet buffer control is corrupted, the receive path will be unusable until it is reset.

Additionally, when running Linux, an invalid descriptor that indicates a packet length larger than the MTU may cause a kernel panic..

Workaround: Implement the following modifications to the initialization sequence:

1. Before setting RCTL.RXEN:
   - Clear RXDCTL.ENABLE for all queues that are in use.
   - Write 0x0000 to RLPML.LPML.
   - Set RCTL.LPE and clear RCTL.SBP.
   - Clear RFCTL.LEF.
2. Set RCTL.RXEN.
3. Wait 2 ms.
4. Restore RXDCTL, RLPML, RCTL, and RFCTL to the intended initial values.
5. (Optional) Read to clear ROC, RNBC, and MPC since these statistics counters may have been incremented during the previous steps.

Status: No Fix. There are no plans to fix this erratum. However, Intel drivers will implement the SW workaround.

18. PCIe: Reception of Completion That Should Be Dropped, Can Occasionally Cause Device Hang or Data Corruption.

Problem: This erratum can occur when the 82575 PCIe receives a completion that should be dropped, while the 82575 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 82575 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 82575 occasionally responds with a NAK as a result of noise on the line. This NAK could cause a completion to be replayed. The 82575 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 82575 was starting to send. The type and result are listed below:

- TX descriptor with TSO – 82575 offload machine may hang.
- TX data or Tx descriptor without offload – 82575 will transmit a packet on the network with invalid data but a valid CRC
- RX descriptor – 82575 will DMA a receive packet to the wrong Memory address.

Workaround: Disabling L0s in the switch port to which the 82575 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; there are no plans to fix this erratum.

7.0 Sightings

(None.)

8.0 Specification Clarifications

1. PCIe partial Memory Write requests are actually writing the full DW.

Clarification: The PCIe specification allows a device to not accept certain requests. This is under the "programming model" case. The device needs to issue a Completer Abort error if the specific request violates the programming model. As part of its programming model, the device does not support writes and reads with Byte enables to specific memory addresses. Such writes will be full executed and will not be treated as completion abort.

CSR writes and reads with partial (or zero) Bytes Enables will be executed (In specific address ranges). This scenario will not happen 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 or zero bytes enable writing to the device.

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

This page intentionally left blank.