Intel® 810E Chipset Family: 82810E Graphics and Memory Controller Hub (GMCH)

Specification Update

June 2001

Notice: The Intel® 82810E GMCH may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are documented in this Specification Update.

Order Number: 290694--003

Contents

Revision History

Rev.	Draft/Changes	Date
-001	Initial Release; Errata #1-12	January 2001
-003	Information from NDA –002 and –003 included in this release. Added information to the Component Marking Information table	June 2001

Preface

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

Information types defined in the Nomenclature section of this document are consolidated into this update document and are no longer published in other documents. This document may also contain information that has not been previously published.

Affected Documents/Related Documents

Document Title	Document Number
Intel® 810E Chipset Family: 82810E Graphics and Memory Controller Hub (GMCH) datasheet	290676-002

Nomenclature

Specification Changes are modifications to the current published specifications. These changes will be incorporated in the next release of the specifications.

Errata are design defects or errors. Errata may cause the Intel® 82810E 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 describe a specification in greater detail or further highlight a specification's impact to a complex design situation. These clarifications will be incorporated in the next release of the specifications.

Documentation Changes include typos, errors, or omissions from the current published specifications. These changes will be incorporated in the next release of the specifications.

Component Identification via Programming Interface

The Intel® 82810E GMCH may be identified by the following register contents:

Stepping	Vendor ID[1]	Device ID[2]	Revision Number[3]
A3	8086h	7124h/7125h	03h

NOTES:
1. The Vendor ID corresponds to bits 15-0 of the Vendor ID Register located at offset 00-01h in the PCI function 0 configuration space.
2. The Device ID corresponds to bits 15-0 of the Device ID Register located at offset 02-03h in the PCI function 0 configuration space.
3. The Revision Number corresponds to bits 7-0 of the Revision ID Register located at offset 08h in the PCI function 0 configuration space.

Component Marking Information

The Intel 82810E GMCH may be identified by the following component markings:

Stepping	S-Spec	Top Marking	Notes
A3	SL3MD	FW82810E	Production 82810E GMCH
A3	SL3P6	FW82810DC100	Production 82810DC100 GMCH
A3	SL3P7	FW82810	Production 82810 GMCH

Summary Table of Changes

The following table indicates the Specification Changes, Errata, Specification Clarifications or Documentation Changes, which apply to the listed Intel 82810E GMCH steppings. Intel intends to fix some of the errata in a future stepping of the component and to account for the other outstanding issues through documentation or Specification Changes as noted. This table uses the following notations:

Codes Used in Summary Table

X:	Erratum, Specification Change or 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.
(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.

Number	SPECIFICATION CHANGES
	There are no specification changes in this Specification Update revision

Number	Steppings		Plans	ERRATA
	A0			
1	X		Fix	Host Interface RCOMP
2	X		NoFix	Overlay TLB
3	X		NoFix	Flat Panel Pixel Doubling Mode
4	X		NoFix	Anisotropic Texture Mapping
5	X		NoFix	HAB[7] Driven
6	X		NoFix	3D Texture Color/Chroma Key
7	X		NoFix	Data Decoded As Command during Low Priority to Interrupt Priority Ring Buffer Transition
8	X		NoFix	Video Overlay Bandwidth
9	X		NoFix	AC97 Latency and Drop Out
10	X		NoFix	Asynchronous Queue Overflow
11	X		NoFix	PM_CS Power State Bits Accept Invalid States
12	X		NoFix	Asynchronous Screen Flip

Number	SPECIFICATION CLARIFICATIONS
	There are no specification clarifications in this Specification Update revision

Number	DOCUMENTATION CHANGES
	There are no documentation changes in this Specification Update revision

Specification Changes

There are no specification changes in this Specification Update revision.

This page is intentionally left blank.

Errata

1. Host Interface RCOMP

Problem: Depending upon system design and environmental factors, a violation of the host interface hold timing specification may occur. Root cause is a circuit (RCOMP) designed to dynamically adjust clock to out delay (Tco) and slew rate on host interface buffers that does not function.

Implication: If hold timing specifications are violated, data corruption may occur. This data corruption may also result in a system hang.

Workaround: A BIOS workaround is available to fix the Tco and Slew Rate at optimum values.

Status: Planned fix in next stepping of silicon. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

2. Overlay TLB

Problem: The GMCH does not correctly determine when it is valid to throw away the old translations in the TBL and replace them with a new set of translations when the following conditions are met
-Overlay is operating in 4:2:0 or 4:1:0 modes
-The overlay surface is read from linear memory
-An aligned 8 KB boundary falls in the middle of a scanline
-Either X-mirroring or Y-mirroring is activated

Implication: If the Conditions stated above are met the GMCH will display incorrect data on the screen resulting in some lines of color on the screen.

Workaround: Do not run overlay in 4:2:0 or 4:1:0 modes when the surface is in linear memory and either horizontal or vertical mirroring is turned on.

Status: No Stepping Fix. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

3. Flat Panel Pixel Doubling Mode

Problem: The flat panel display engine change cannot handle mode switches into a flat panel pixel doubling mode without being disabled first.

Implication: If an application changes to a pixel doubling mode the entire screen will display the border color.

Workaround: The work around requires the BIOS to detect a write to the pixel doubling bit (SR01[3]). When a write to this bit is detected, the flat panel engine must be disabled and restarted. This results in proper DCLK/FCLKOUT synchronization. Windows does not allow applications to write to registers that set pixel doubling modes. Legacy DOS applications that attempt to change the video mode to pixel doubling mode can still cause this issue to appear.

Status: BIOS workaround, No planned stepping fix. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

4. Anisotropic Texture Mapping

Problem: Anisotropic logic with multiple textures uses the incorrect texel under certain conditions with color or chroma keying enabled. This issue takes place during texturing in the case of a multi-texture polygon, when the first texture is anisotropic and the second is not, and the first texture has color or chroma key on. In this case the kill bit for the last pixel of a 4X4 pixel span for the first texture can get lost and the pixel is displayed when it should not. This problem requires a unique set of conditions to appear. First, anisotropic logic must be turned on, which requires more video processing in the video hardware. Second, multiple textures must be applied to a single polygon.

Implication: This will cause a pixel to display incorrectly, which may result in a sparkle on the screen.

Workaround: None

Status: No planned fix. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

5. HAB[7] Driven

Problem: When the IOQ depth is set to one, a test mode, HAB[7] is driven to Vil until ADS# asserts on reset. When operating correctly the GMCH should float HAB[7] after two clock cycles following the deassertion of RESET#.

Implication: This problem does not affect the normal operation of Intel® 810 chipset platforms, which sets the IOQ depth at four. No contention on the host address bus results from this issue.

Workaround: Set the IOQ depth to four for normal operation. Do not set the IOQ depth to one, except for validation purposes.

Status: No planned fix. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

6. 3D Texture Color/Chroma Key

Problem: A case exists where invalid data from the texture cache is included into the bilinear filter to determine the texture color for a pixel causing the shading for that pixel to be incorrect. This occurs when an indexed texture map has color or chroma key enabled and the filtering appears on a texture boundary. In this case one valid texel is keyed on and the other is keyed out. In this case the bilinear filtering uses two invalid texels in the cache that are past the texture boundary. If the valid texel that is keyed out lines up with an invalid texel that is keyed on, the valid texel, which is keyed out, is incorrectly replaced with the invalid texel before the bilinear filtering.

Implication: In this case a pixel may be shaded incorrectly on the edge of a texture.

Workaround: None.

Status: No planned fix. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

7. Data Decoded As Command during Low Priority to Interrupt Priority Ring Buffer Transition

Problem: Data is decoded as a command packet if the low priority ring is executing non-pipelined state variable packets and the interrupt priority ring is enabled. This could happen anytime 3D is active and a stretch-blit (2D) operation is conducted through the interrupt ring.

Implication: The result could be a hang of the parser state machine or incorrect setting of the state variable in 3D. The only known application that uses the interrupt priority ring is the stretch-blit operation. Intel drivers associated with the 82810E do not use this process. This process is used by DVD software.

Workaround: Do not use non-pipelined state variable execution when the interrupt ring is enabled. DVD software writers are aware of this issue and write software correctly.

Status: There are no plans to fix this erratum in silicon. Software fixes are already in place. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

8. Video Overlay Bandwidth

Problem: Bandwidth limitations impact the use of graphics modes of 1152X864X24 @ 85 Hz and at 1280X1024X24 at 85 Hz with AVI or JPG video files.

Implication: Graphics corruption such as vertical stripes may appear while moving around an active AVI or JPG video overlay window.

Workaround: None identified. Using CAS latency 2 memory reduces the problem.

Status: There are no plans to fix this erratum in silicon. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

9. AC97 Latency and Drop Out

Problem: Cycles from the local cache fill up the command queue causing aperture cycles from the host to be starved. AC97 cycles are queued with a lower priority than the aperture cycles and can therefore experience under-run.

Implication: Latency or data drop out may occur in the completion of AC97 cycles. Audio cycles may not be synchronized with video. MODEM operation may drop out entirely during heavily loaded system operation. These effects may vary by application, user, OS in use, and system load.

Workaround: None identified.

Status: There are no plans to fix this erratum in silicon. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

10. Asynchronous Queue Overflow

Problem: A specific heavily loaded system configuration and specific traffic causes the 82810E GMCH asynchronous queues to fill up, causing up-bound I/O traffic to get blocked while waiting for an I/O transaction to complete. If the following specific configuration and transaction sequence occurs, the system may hang:

1. A processor memory write to the hub interface occurs,
2. AND a QWord misaligned processor read to PCI or LPC occurs,
3. AND any three of the following four interfaces are simultaneously active,
   (1) PHLD traffic from M-ISA or LPC to DRAM,
   (2) IDE BM traffic to DRAM,
   (3) PCI Master #1 read traffic from DRAM,
   (4) PCI Master #2 read traffic from DRAM,
4. AND sufficient system traffic exists to fill the asynchronous upbound and downbound queues in the GMCH.

Implication: If the specific configuration and transaction sequence shown above occurs, the system may hang.

Workaround: Disable PCI pre-fetching in the ICH. This may cause a 1%-2% performance hit on PCI initiated reads to DRAM only.

Status: There are no plans to fix this erratum in silicon. A software fix to disable PCI pre-fetching will be published. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

11. PM_CS Power State Bits Accept Invalid States

Problem: PCI Power Management Control/Status Register (PM_CS), Device 1, address offset E0h-E1h, bits [1:0], accepts values representing power management states D1 and D2, which the hardware device does not support.

Implication: This is a violation of the PM 1.1 specification and causes the WHQL PC99A HCT9.x test to fail.

Workaround: None

Status: This issue will not be fixed in the 82810E GMCH. Intel is working with Microsoft on a WHQL waiver for WHQL certification. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

12. Asynchronous Screen Flip

Problem: When the Intel® 82810E device is configured for asynchronous screen flipping, under certain timing-dependent circumstances the display engine may temporarily read pixel data from a random memory location.

Implication: When changing display surfaces using the asynchronous screen flipping, subtle display corruption is seen in the form of short, somewhat random colored, horizontal lines along the left side of the screen.

Workaround: Driver version 4.1.1 does, and future version will, disable asynchronous screen flipping for commonly used 3D resolutions.

Status: There are no plans to fix this erratum in silicon. Refer to the *Summary Table of Changes* to determine affected product(s) and stepping(s).

This page is intentionally left blank.

Specification Clarifications

There are no specification clarifications in this Specification Update revision.

Documentation Changes

There are no documentation changes in this Specification Update revision.