Intel® 82443MX PCIset

Specification Update

February 2003

Notice: The Intel® 82443MX PCIset 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.

Document number: 245051-008

Contents

Revision History

Date of Revision	Revision	Description
6/22/99	-001	Document includes all known specifications to date (original version).
9/18/99	-002	Added Erratum #32 Added workarounds to Errata #30 and #31
01/18/99	-003	Added Erratum #33 Added Document Changes #1, #2 and #3
08/18/00	-004	Added Erratum #34 and #35
03/06/01	-005	Added Erratum #36 Updated Errata #33 - Removed workaround option #3 (Option #3 will be revised in Q2 2001 release of 440MX Specification Update for USB high-speed device compatibility.)
06/7/01	-006	Update to Errata #33 – Workaround Option #3 revised; added Workaround Option #4 Removed "GTL VREF Document Correction for Intel® 82443MX PCISet Electrical and Thermal Specification" in Document Changes. See Intel® 82443MX PCISet Datasheet Addendum "Electrical and Thermal Specification" on developer.intel.com for updated GTL VREF.
1/7/02	-007	Added documentation changes #3-5: Compatibility for mobile Pentium® III processor-M
3/10/03	-008	Added documentation change #6 – NAND Tree Testability

Preface

This document is an update to the specifications contained in the Intel® 82443MX PCIset Datasheet and other documents related to the 82443MX PCIset. It is intended for hardware system manufacturers. It contains Specification Changes, Errata, Specification Clarifications, and Documentation Changes.

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 82443MX PCIset 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.

Note: Errata remain in the specification update throughout the product's lifecycle or until a particular stepping is no longer commercially available. Under these circumstances, errata removed from the specification update are archived and available upon request. Specification changes, specification clarifications, and documentation changes are removed from the specification update when the appropriate changes are made to the appropriate product specification or documentation.

Summary Table of Changes

The following table indicates the Specification Changes, Errata, Specification Clarifications or Documentation Changes that apply to the listed Intel 82443MX PCIset 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.

Errata

Number	Stepping(s)		Page	Status	Errata
	A-0	B-0			
1	X		10	Fixed	AC'97 Multifunction Configuration not recognized
2	X		10	Fixed	AC'97 Dynamic Clock Gating corrupts data
3	X		11	Fixed	AC'97 Cold Reset does not clear Codec Access Semaphore Bit and Read Completion Bit
4	X		11	Fixed	SMI/SCI for Wake-on-Ring not generated in Dual Codec System
5	X		11	Fixed	SDRAM Leadoff Command Timing must be set to 4 CPU Clocks
6	X	X	11	NoFix[1]	C3 Power State/BMIDE and Type-F DMA Livelock
7	X	X	12	NoFix	Device Monitor 9 and accesses to IO locations 62/66h
8	X		12	Fixed[2]	USB Port Enable and Post Status Bits
9	X		12	Fixed	C3/POS/STR Memory Hang condition
10	X		14	Fixed	SUSTAT1# Implemented as SUSTAT#
11	X	X	14	No Fix	ISA Verify followed by PCPCI DMA
12	X		15	Fixed	USB Dribble
13	X		15	Fixed	IDE PREFETCH
14	X	X	16	NoFix	PCI accesses to External PCI-based IDE Devices will not cause Power Management events

Number	Stepping(s)		Page	Status	Errata
	A-0	B-0			
15	X	X	16	NoFix	Burst Events may cause LVL2 or LVL3 register reads to be missed
16	X	X	17	NoFix	Device Trap
17	X	X	17	NoFix	Daylight Savings Time Errata
18	X		18	Fixed	Native Audio Bus Master Base Address
19	X		18	Fixed	Native Audio Bus Master Base Address
20	X		18	Fixed	AC_RSTB Errata
21	X	X	18	NoFix[2]	Boundary Condition issues for USB Connects and Disconnects
22	X		20	Fixed	Incorrect Data transmitted through AC'97 Modem Out Channel
23	X		20	Fixed	Port Enable/Disable Change Bit incorrectly Set for OC# Assertion During S1-S5 States
24	X	X	20	NoFix	Minimum latency to access USB I/O Registers after exiting from Suspend
25	X		21	Fixed	USB Transaction initiated without Ample Time to complete
26	X		21	Fixed	Mismatched PIDs Lead to False Bus Cycle
27	X	X	21	NoFix	Native Modem Bus Master Base Address
28	X	X	22	NoFix	AC'97 PCM Out
29		X	22	NoFix	Incorrect Recognition of High/Low speed USB device out of Resume
30		X	23	NoFix	Power Management General Purpose Status Register "USB_STS" bit not Cleared Correctly
31		X	23	NoFix	USB Port Status and Control Register "Connect Status Change" bit Incorrectly Set
32	X	X	24	NoFix	USB Rise/Fall Time Matching
33	X	X	24	NoFix	USB Connect Status Change Bit does not get set under certain conditions.
34		X	27	NoFix	AC97 Soft Audio and Soft Modem Issue during Intel® SpeedStep® Technology Operating Mode Transitions
35		X	27	NoFix	GPIO22/PIRQC# and GPIO23/PIRQD# Errata
36		X	27	NoFix	AC'97 Master Abort Errata

NOTE: This erratum will be fixed only for BMIDE in B-0 stepping. Fixed for certain boundary conditions. See Erratum #21 for details.

Specification Changes

Number	Stepping(s)		Page	Status	Specification Changes
	A-0	B-0			
N/A		X	30		None in this specification update revision

Specification Clarifications

Number	Page	Specification Clarifications
N/A	31	None in this specification update revision

Documentation Changes

Number	Document Revision	Page	Status	Documentation Changes
1		32		PME# Signal Description Correction
2		32		Thermal Design Power Change for 66-MHz Operation
3	Intel® 440MX Datasheet Addendum (Order #273502-001)	32		440MX Processor Side Bus Vref
4	Intel® 440MX Datasheet Addendum (Order #273502-001)	33		440MX Processor Side Bus V_{TT}
5	Intel® 440MX Datasheet Addendum (Order #273502-001)	33		440MX Processor Side Bus Undershoot/Overshoot
6		34		NAND Tree Testability

Identification Information

This section covers the Intel® 82443MX PCIset.

Intel 82443MX PCIset Marking Information

Stepping	Host Bus/SDRAM Frequency	Package	Top Marking	Notes
A-0	66	BGA	FW82443MX Q702ES	Engineering Sample
B-0	66	BGA	FW82443MX Q728ES	Engineering Sample
B-0	66	BGA	FW82443MX SL37L	Production Units
B-0	100	BGA	FW82443MX100 SL3N4	Production Units

Errata

1. Multifunction Configuration Not Recognized

Issue: AC'97 Audio/Modem PCI Functions can not be seen by the operating system because the chipset read-only multi-function bit (bit 7) in the Header Type Register (Device 0, Function 0, Register 0Eh) is incorrectly set to "0." Since a setting of "0" indicates a single function device, operating systems are unable to recognize both audio and modem PCI functions.

Implication: Operating system drivers will not enable audio/modem functionality through AC Link when this bit is set to "0."

Note: The 440MX systems using PCI Audio/Modem are not affected.

Workaround: The BIOS must alter the setting of the multifunction bit in the Header Type Register to return "80h" when read. This will allow AC'97 function 1 and function 2 to be assessable when the OS or driver calls those PCI BIOS functions. The following three PCI BIOS function calls need to be modified in both 16-bit and 32-bit services:

1. ax=B108: Read_Config_byte
2. ax=B109: Read_Config_word
3. ax=B10A: Read_Config_dword

Status: Fixed in B-0 stepping.

2. AC'97 Dynamic Clock Gating Corrupts Data

Background: The AC-link protocol provides for a special 16-bit time slot (Slot 0) wherein each bit conveys a valid tag for its corresponding time slot within the current audio frame. A "1" in a given bit position of slot 0 indicates that the corresponding time slot within the current audio frame has been assigned to a data stream and contains valid data. If a slot is "tagged" invalid, then it is the responsibility of the source of the data (AC'97 for the input stream and AC'97 controller for the output stream) to stuff all bit positions with "0's" during that slot's active time. SYNC remains high for a total duration of 16 BIT_CLKs at the beginning of each audio frame. The portion of the audio frame where SYNC is high is defined as the "Tag Phase." The remainder of the audio frame where SYNC is low is defined as the "Data Phase."

Issue: When the power management feature dynamic clock gating is enabled, outgoing AC'97 data slots 3, 4, and 5 will be corrupted. Dynamic clock gating is enabled by setting the GCLKEN bit in the Power Management Control Register (Device 0, Function 0, Register 7Ah) to a "1." The operation of the other functions remains unaffected.

Implication: The 440MX based platforms using audio, modem, or audio/modem through the AC Link will be affected with data corruption in slots 3, 4, and 5. Designs using PCI based audio, modem, or audio/modem are unaffected.

Workaround: The BIOS must disable the Gated Clock Enable (GCLKEN), bit 2, in the Power Management Control Register (PMCR).

Status: This erratum only affects stepping A0 and will be fixed in stepping B0.

3. AC’97 Cold Reset Does Not Clear Codec Access Semaphore Bit and Read Completion Bit

Issue: An AC’97 Cold Reset does not clear the Codec Access Semaphore bit and Read Completion bit. The Cold Reset bit is located at bit 1 of the Global Control, the Codec Access Semaphore bit is located at bit 0 of the Codec Access Semaphore Register, and the Read Completion bit is located at bit 15 of the Global Status Register.

Implication: The AC’97 driver will not be able to use the AC’97 controller if it thinks the Codec is busy with the Semaphore bit set.

Workaround: Do not issue a Cold Reset command when the Codec Access Semaphore bit or the Read Completion bit is set. All I/O transactions should be completed before attempting to issue a cold reset. The AC’97 controller will clear the semaphore bit upon completion of the I/O transaction. The AC’97 driver should also clear the Read Completion bit before issuing a cold reset.

Status: This erratum only affects stepping A0 and will be fixed in stepping B0.

4. SMI/SCI for Wake-on-Ring Not Generated in Dual Codec System

Issue: In a dual codec system, the 440MX will not generate a SMI/SCI for Wake-on-Ring required to load the modem driver (which has been unloaded when the modem codec transitioned to the D3 state) when audio codec is alive (and therefore AC Link is active).

Implication: This prevents proper wakeup via AC’97 protocol.

Workaround: None.

Status: Fixed in B-0 stepping.

5. SDRAM Leadoff Command Timing Must Be Set to Four CPU Clocks

Issue: Timing issues exist when the Leadoff Command Timing (LCT) bit, bit 3 in the SDRAM Control Register (Device 0, Function 0, Register 76-77h), is programmed to a “1.” This corresponds to a CS# leadoff time of three CPU clocks.

Implication: A CS# leadoff time of 3 is not allowed.

Workaround: Program the LCT bit to “0”, which corresponds to a CS# leadoff time of four CPU clocks.

Status: Fixed in B-0 stepping.

6. C3 Power State/BMIDE and Type-F DMA Livelock

Issue: The 440MX will not always correctly reflect BMIDE and Type-F DMA activity on the BMSTS bit in the Power Management Status Register (PMSTS) of the 440MX Function 3.

Implication: When the OS enters a C3 state, it will disable the arbiter and then perform a PLVL3 register read to enter the C3 state, causing LIVELOCK to occur and resulting in a system hang.

Workaround: In the OS initialization code, DISABLE Type-F DMA if BIOS indicates C3 support. If BIOS indicates that C3 is not supported, leave Type-F DMA enabled. Note that this workaround is only for Type-F DMA and that there is no workaround for BMIDE.

Status: This erratum will be fixed for BMIDE in B-0 stepping. The workaround must be implemented for Type-F DMA.

Note: This erratum was carried over from the Intel® 82371EB (PIIX4E).

7. Device Monitor 9 and Accesses to IO Locations 62/66h

Issue: If Device 9 Idle Enable (IDL_EN_DEV9), Burst Reload Enable (BRLD_EN_DEV9), or Global Reload Enable (GRLD_EN_DEV9) bits are set, then the idle, burst, or global standby timer will reload for I/O accesses to ISA Legacy addresses 62 or 66h. This is regardless of the Generic Decode Monitor Enable bit setting (GDEC_MON_DEV9).

If Device 9 Trap Enable bit (TRP_EN_DEV9) is set, the 440MX generates a trap SMI for accesses to ISA Legacy addresses 62 or 66h. This is regardless of the Generic Decode Monitor Enable bit setting (GDEC_MON_DEV9) and the value of the Programmable Base Address and Programmable Mask register settings (BASE_DEV9 and MASK_DEV9).

Implication: Device 9 cannot be used as a monitor for I/O device addresses exclusive of 62 and 66h. GPI4 can not be used exclusively to reload the idle, burst, or global standby timers because accesses to ISA Legacy addresses 62 or 66h will also reload the times.

Note: GPI4 is still available as a General Purpose Input.

Workaround: None. If a generic I/O device monitor exclusive of I/O address 62 and 66h is needed, then use Device 10 if it is available.

Status: This will not be fixed in future steppings of the 440MX.

Note: This erratum was carried over from the *Intel® 82371EB (PIIX4E).*

8. USB Port Enable and Post Status Bits

Issue: A boundary condition can occur while the 440MX is entering a POSCL, STR, STD, or SOFF suspend state and a USB wake event occurs. When the 440MX resumes from POSCL, STR, STD, or SOFF the port enable and port status bits are no longer set. These bits are in the Port Status and Control Register (PORTSC).

Implication: The OS or application may no longer be able to use USB.

Workaround: None.

Status: Fixed in B-0 stepping.

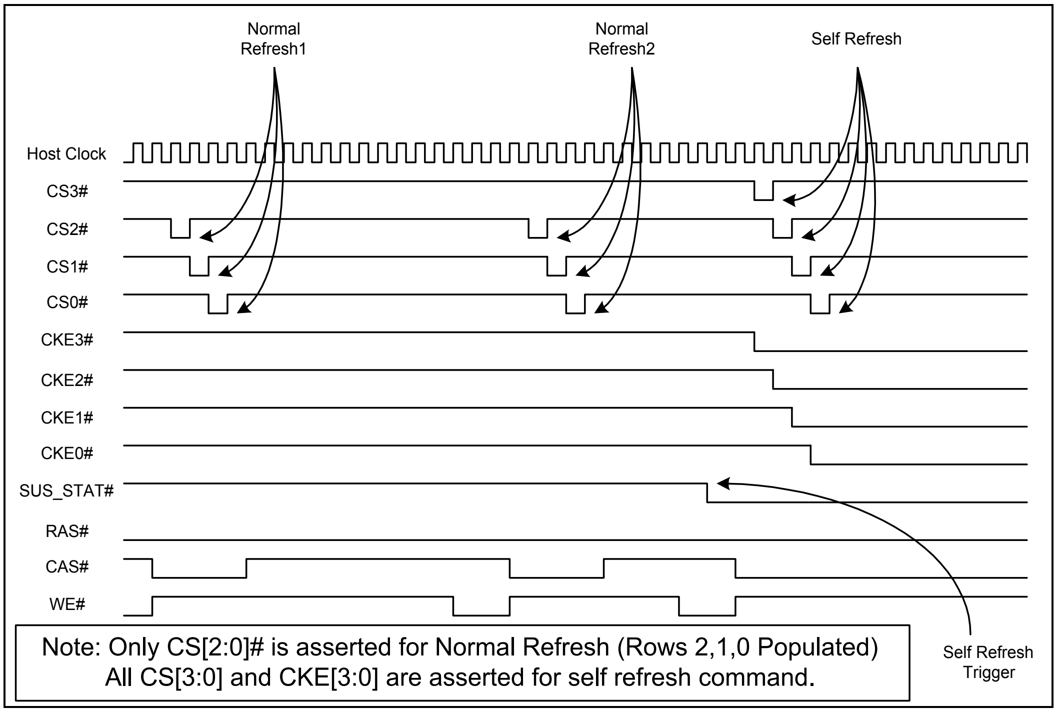
9. C3/POS/STR Memory Hang Condition

Issue: SDRAM may be prevented from being put into self-refresh state when a self-refresh entry request triggered by SUS_STAT# assertion collides with a normal refresh caused by internal timer expiration. The collision between the two requests results in anomalous behavior on the CS[0-3]# and CKE[0-4]# , which can cause the system hang/memory corruption. There is a three Host clock window in which the assertion of SUS_STAT# can cause this failure.

The following describes both a successful self-refresh entry and an unsuccessful self-refresh entry.

Figure 1 illustrates a successful suspend entry sequence for a system which has rows 0, 1, and 2 populated. Note that for normal refresh, only the chip selects for the populated rows are asserted, whereas for the self-entry command, all four chip selects are asserted regardless of its population. Assume that dynamic SDRAM power-down feature is disabled. If all four rows are populated, it is impossible to capture this erratum.

Figure 1: Successful Self-refresh Entry

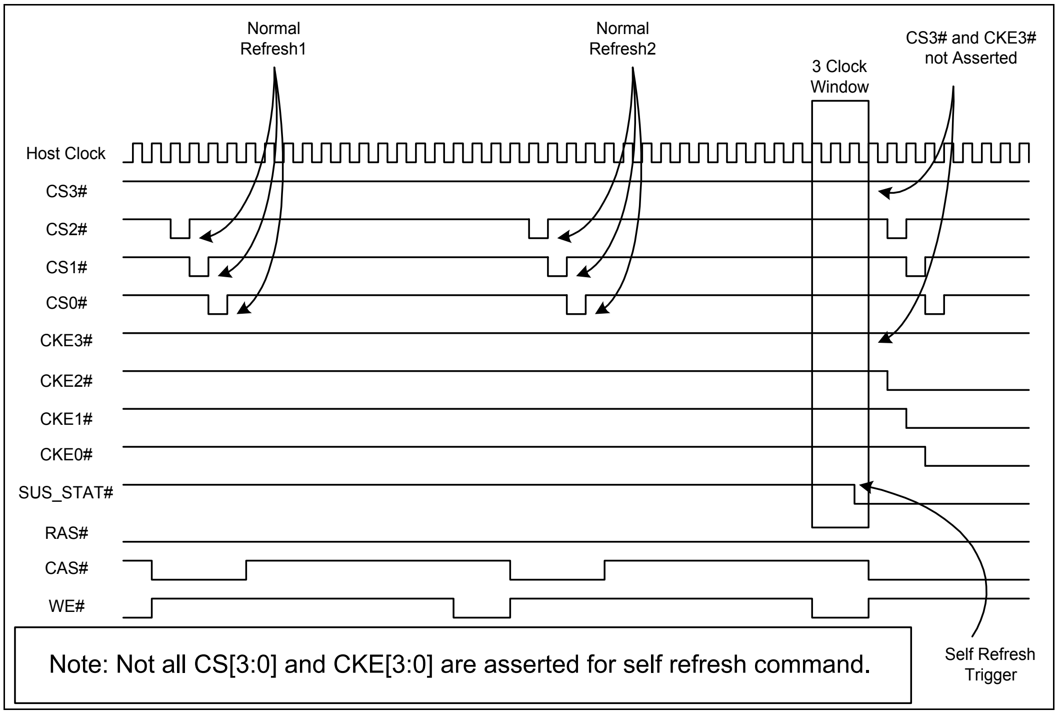


An unsuccessful self-refresh will exhibit the following characteristics:

1. SUS_STAT# assertion is taking place around a normal refresh request.
2. CS# and CKE# signals are asserted only for the populated rows.

Figure 2 illustrates an unsuccessful self-refresh entry sequence. Note that SUS_STAT# is being asserted approximately the same time that the third normal refresh would have taken place. Since the SUS_STAT# assertion hits the three host clock window, CS3# and CKE3# do not get asserted during the self-refresh entry sequence.

Figure 2: Unsuccessful Self-refresh Entry



Implication: The system may hang if an unsuccessful self refresh entry sequence occurs.

Workaround: Please refer to the SDRAM Suspend Refresh Erratum APM Workaround Write-up, Rev 1.0 for a software workaround solution.

Status: Fixed in B-0 stepping.

Note: This erratum is was carried over from the *Intel® 82443BX AGPset*.

10. SUSTAT# Implemented Incorrectly

Issue: SUSTAT# normally notifies peripheral devices (Graphic or Cardbus controllers) during POS, POSCCL, POSCL, or STR states. The current implementation of SUSTAT# notifies peripheral devices during POS, POSCCL, POSCL, STR, or C3 states.

Implication: SUSTAT# will be incorrectly asserted in C3.

Workaround: Please contact Intel for partial solutions.

Status: Fixed in B-0 stepping.

11. ISA Verify Followed by PCPCI DMA

Issue: Upon completion of an ISA Verify Mode cycle that reaches Terminal Count (TC), the 440MX will not transition an internal TC signal from the TC state to the Idle state.

Implication: If a PCPCI DMA cycle follows an ISA DMA Verify cycle that reaches terminal count with no other DMA, ISA Master, or ISA Refresh cycles between them, then the 440MX will assert the TC signal on the first data transfer of the PCPCI DMA cycle. This results in an incomplete data transfer.

Workaround: None.

Status: This will not be fixed in future steppings of the 440MX.

Note: This erratum was carried over from the *Intel® 82371EB (PIIX4E).*

12. USB Dribble

Issue: A USB receive packet with a bitstuff following the transmission of CRC, coupled with a dribble bit due to prop delays through cables and HUBs, may be incorrectly interpreted by the USB host controller state machine as a poorly formed EOP.

Implication: The host controller response to this is a non-acknowledge with a CRC/Timeout status communicated to the software. If this condition persists, the error count associated with this packet will be exceeded and an interrupt can be generated to software. This will stall the USB device. Current software reports a device error to the user via a pop-up window. Another implication is that the installed base may have limited USB expandability via HUBs.

Workaround: There are two possible workarounds.

1. Hardware: Attach the USB device into a USB port closer to the root hub.
2. Software: Detect the CRC/Timeout error and count exceeded and attempt to re-queue the packets while changing the length of the packets. Changing the length of the packets will change the CRC and thus will likely remove the combination of the two events causing the failure.

Status: Fixed in B-0 stepping.

Note: This erratum was carried over from the *Intel® 82371EB (PIIX4E*).

13. IDE Prefetch

Issue: While executing a PIO IDE Read Sector(s) or Read Multiple command with PIO pre-fetching enabled, a read of a Non-data Register (such as ALT STATUS Register) may cause the 440MX PIO pre-fetch counter to increment, incorrectly since it should only increment on data transfers.

Implication: Invalid data may be written to memory. Due to Intel customers should perform their own risk analysis on this errata and determine the most appropriate work around for their systems.

Workaround: Three possible workarounds exist:

1. Do not perform Non-data register reads while an IDE PIO transfer is taking place. In cases where this erratum has been seen, an interrupt (IRQn or SMI) has been used to enter the code from which the ALT STATUS read occurs. Code that is not directly involved in the IDE transfer should not perform the ALT STATUS read to check status of IDE transfers.
2. Use the IDE device idle timer to detect IDE activity.
3. Disable IDE PIO prefetching.

Status: Fixed in B-0 stepping.

Note: This erratum was carried over from the *Intel® 82371EB (PIIX4E).*

14. PCI Accesses to External PCI-based IDE Devices Will Not Cause Power Management Events

Issue: PCI accesses to external IDE devices on the PCI bus do not generate power management events (Idle timer reloads, global standby timer reloads, burst timer reloads, and I/O traps).

Implication: Power management of external PCI-based IDE devices must use other means to monitor the activity of those devices.

Workaround: System BIOS should use the following methods to monitor external PCI-based IDE devices:

1. If there is a need to monitor accesses to the IDE controller to keep the global standby timer from expiring, the IRQs should be enabled (GRLD_EN_IRQ) as a reload event for the global standby timer.
2. If there is a need to monitor an external IDE controller for idleness, use the following algorithm:
   a. Disable the external IDE controller. Set the 440MX to trap on the IDE access and enable the internal IDE controller.
   b. When the SMI is generated, the idle timer can be started, the internal IDE controller disabled, and the instruction redone to the external IDE controller. The IDE device is then assumed to be active during idle timer count down.
   c. When the idle timer times out, an SMI is generated and the 440MX should again be set to trap, the external IDE device disabled, and the idle timer started.
   d. If the idle timer times out before the trap occurs, then the external IDE controller is idle and can be put into a lower power mode. The 440MX is then set up to trap as in step 3 below.
   e. If the trap occurs first, the IDE device is not idle. The BIOS then returns to step b above.
3. If there is a need to perform I/O trapping on an external IDE controller, set the 440MX to trap on the IDE access and enable 440MX internal IDE controller. When the SMI is generated, the internal IDE controller can be disabled, the external controller enabled, and the I/O cycle restarted.

Status: This will not be fixed in future steppings of the 440MX.

Note: This erratum was carried over from the *Intel® 82371EB (PIIX4E).*

15. Bus Master IDE Timeout

Issue: During an IDE DMA write, the 440MX IDE controller will invalidate its FIFO if the IDE device deasserts its DREQ signal for greater than 1 us. During the FIFO invalidation, the 440MX does not prevent an FIFO fill from PCI.

Implication: In Bus Master IDE (BMIDE) mode, the PCI interface is prefetching data. If this prefetched data gets inserted into the IDE FIFO (during a FIFO invalidation due to DREQ deassertion > 1 us) the IDE controller will lock up. Any future reassertion of the DREQ signal will not be acknowledged by the 440MX IDE controller. BMIDE transactions will not complete on either the primary or secondary channel.

Workaround: If the controller locks up, the BMIDE driver must timeout, reset 440MX Start/Stop Bus Master bit, and retry the transfer. Note that this erratum does not occur using PIO mode or Ultra DMA/33 mode.

Status: This will not be fixed in future steppings of the 440MX.

Note: This erratum was carried over from the *Intel® 82371EB (PIIX4E).*

16. Burst Events May Cause LVL2 or LVL3 Register Reads to Be Missed

Issue: Burst events that occur after Burst Enable bit (BST_EN) has been set and before the Processor Level 2 (LVL2) or Processor Level 3 (LVL3) register read may cause the LVL2 or LVL3 read to be missed.

Implication: When the above conditions occur, the system will not transition into the Level 2 or Level 3 clock control condition as intended but will remain at full speed.

Workaround: Software must ensure that no external burst events are active when placing the system into a LVL2 or LVL3 state. To ensure this, prior to LVL2 or LVL3 register read, only the Device 3 idle timer should be enabled as a burst event. The device 3 idle timer is then enabled with all reload events disabled. The LVL2 or LVL3 register read is performed placing the system into a LVL2 or LVL3 clock control condition. The Device 3 idle timer will then generate a burst event upon expiration. During this first burst, the desired burst events are then enabled. The system then functions as expected.

Status: This will not be fixed in future steppings of the 440MX.

Note: This erratum was carried over from the *Intel® 82371EB (PIIX4E).*

17. Device Trap

Issue: When the 440MX has the Device Trap logic enabled for devices 0,1, and 4-13, it forwards the I/O access cycles for the device to the EIO/ISA and IDE Bus.

Implication: Accesses to devices in a powered-down state could cause unpredictable results.

Workaround: Upon a power-down event for devices 0 and 1 (IDE) the SMI handler must save the IDE register settings in CMOS, disable IORDY, and set PIO transfers for compatible timings. Upon a power-up event for devices 0 and 1, the SMI handler must restore all original IDE register settings and re-initialize the IDE drive to the original Mode (PIO mode or DMA).

Upon a power-down event for all other devices (using EIO), the SMI handler must disable the EIO decode and enable the trap logic for that device. Upon a power-up event, the SMI handler must enable the EIO decode and disable the trap logic.

Status: This will not be fixed in future steppings of the 440MX.

Note: This erratum was carried over from the Intel® 82371EB (PIIX4E).

18. Daylight Savings Time Errata

Issue: If the last Sunday in October is the 30th or the 31st and the daylight savings enable bit is set, the time will not correctly adjust back one hour from 1:59:59 a.m. to 1:00:00 a.m.

Implication: The system time may not be correct after the daylight savings time change. The first manifestation of this will be on October 31st 1999.

Workaround: Three alternative solutions exist:

1. If using Microsoft Windows 95/98 or Windows NT4.0 operating systems, leave the system on and the operating system running at 1:59:59 a.m. on the last Sunday of October. Some operating systems will correctly detect the time change and correct the CMOS time settings.
2. After the daylight savings fallback occurs, change the time manually using either an operating system date/time function or the BIOS setup.
3. Contact your system provider to see if there is a BIOS update available that corrects this condition.

Status: Fixed in the B-0 stepping.

19. Native Audio Bus Master Base Address

Issue: When bit 7 of the Native Audio Bus Mastering Base Address Register is set to 1, accesses to the Native Audio Bus Master Control Registers followed by accesses to the primary codec will only go to the secondary codec.

Implication: Since the primary codec can not be accessed it can appear non-functional and the system audio could stop.

Workaround: The system BIOS must ensure that bit 7 of the Native Audio Bus Mastering Base Address Register is set to 0.

Status: Fixed in the B-0 stepping.

20. AC_RSTB Assertion

Issue: AC_RSTB is asserted when the 440MX is in POS and POSCCL state.

Implication: The codec(s) will get cold-reset when the 440MX enters POS and POSCCL states and will lose all of their programming.

Workaround: The codec(s) have to be reprogrammed after resuming from POS and POSCCL.

Status: Fixed in the B-0 stepping.

21. Boundary Condition Issues for USB Connects and Disconnects

Issue: Boundary conditions can occur while entering an S1-S5 state. When resuming from POSCL, STR, STD, or SOFF the port enable and port status bits are no longer set. These bits are in the Port Status and Control Register (PORTSC).

The following specifies the boundary conditions that are applicable:

1. If a USB device is *disconnected, connected, and re-disconnected* within 32 µsec within a 90 nsec window after the PCISTP# signal while entering suspend (S1-S5), it will unintentionally wake up the system and may result in USB instability.
2. If a USB device is *connected, disconnected, and re-connected* within 32 µsec within a 90 nsec window after the PCISTP# signal while entering suspend (S1-S5), it will unintentionally wake up the system and may result in USB instability.
3. If a USB device is *disconnected, connected, and re-disconnected* that take longer than 32 µsec within a window of STPCLK# - 3 PCICLKs but before the clocks resume, while entering suspend (S1-S5), it will unintentionally wake up the system and may result in USB instability.
4. If a USB device is connected and disconnected, and that take longer than 32 µsec within a window of STPCLK# - 3 PCICLKs but before the clocks resume while entering suspend (S1-S5), it may unintentionally wake up the system and may result in USB instability.
5. If a USB device is permanently disconnected within a window of STPCLk# - 3 PCICLKs but before the clocks resume while entering suspend (S1-S5), it may result in USB instability.
6. If a USB device is permanently connected, within a window of STPCLK# - 3 PCICLKs but before the clocks resume while entering suspend (S1-S5), it may result in USB instability.
7. If a USB device drives a K state resume after the STPCLK# - 3 PCICLKs and before the clocks stop while entering suspend (S1-S5), which may result in USB instability.

Implication: If the conditions above are meet, USB instability may result.

Workaround: None.

Status: The status for each boundary condition is as follows:

1. This will not be fixed in future steppings.
2. This will not be fixed in future steppings.
3. Fixed in the B-0 stepping for the condition before the clocks are suspended, but it will not be fixed while the clocks are stopped.
4. This will not be fixed in future steppings.
5. Fixed in the B-0 stepping.
6. Fixed in the B-0 stepping.
7. Fixed in the B-0 stepping.

22. Incorrect Data Transmitted Through AC'97 Modem Out Channel

Issue: The following sequence of events may cause incorrect data to be sent out on the AC'97 Modem Out Channel:

1. The AC97 Controller reaches the Last Valid Buffer and the Last Valid Index is updated some time after all the data for the current buffer has been fetched but before it has all been transmitted out.
2. The new buffer starts at an odd word location.
3. The AC97 Controller fetches the new Buffer Descriptor and then fetches the data for the new buffer before the data for the previous buffer (which was the last valid buffer) is sent out on the link.

Implication: The AC97controller may send out data from the lower word of the first data fetch in the new buffer instead of the upper word because the buffer is odd word aligned. This problem is seen only on the Modem Out channel resulting in one extra sample being sent out on the link before the new buffer's samples are sent out. Modem drivers that incorporate an error-checking algorithm will see this issue.

Workaround: None.

Status: Fixed in the B-0 stepping.

23. Port Enable/Disable Change Bit Incorrectly Set for OC# Assertion During S1-S5 States

Issue: The following applies to OC# assertion:

1. OC# assertion during S1-S2 suspend states will not set the Port Enable/Disable Change bit (USB Port Status and Control Register, bit 3) after resuming from suspend. However, the port will be disabled and the Port Enabled/Disabled status bit (USB Port Status and Control Register, bit 2) will be cleared to 0. The Overcurrent Active bit and the Overcurrent Indicator bit (USB Port Status and Control Register, bits 11:10) will be set correctly.
2. OC# assertion during S3-S5 suspend states will not disable Port Enable/Disable Change bit after resuming from suspend. However, the Overcurrent Active bit and the Overcurrent Indicator bit (USB Port Status and Control Register, bits 11:10) will be set correctly. Note that the occurrence of this boundary condition should be extremely rare since an overcurrent during STR, STD, or SoftOff is not expected.

Implication: The Port Enable/Disable Change bit may be incorrect if OC# is asserted during S1-S5 states.

Workaround: When exiting from suspend, BIOS should check whether the Overcurrent Active bit and/or Overcurrent Indicator bit is set to "1." If yes, it should take the appropriate actions to respond to an overcurrent event. In addition, if the Overcurrent Indicator bit is set to "1" when exiting from S3-S5, the BIOS should clear the Port Enabled/Disabled status bit to disable the port. The BIOS should not depend on the Port Enable/Disable Change bit to indicate whether the port has experienced an overcurrent event.

Status: Fixed in the B-0 stepping.

24. Minimum Latency to Access USB I/O Registers After Exiting From Suspend

Issue: When resuming from any suspend state in which PCIRST# was asserted (POSCL, STR, STD, or Soft Off), software must wait a minimum of 8 µs after PCIRST# is deasserted before accessing the USB I/O registers.

Implication: Invalid states may be read from the USB registers and writes may cause invalid operations.

Workaround: When resuming from POS and POSCCL, software must wait 8 µs from the deassertion of PCISTP# (which allows the system-wide clocks to restart) before accessing the USB I/O registers.

Status: This will not be fixed in the B-0 stepping.

25. USB Transaction Intimated Without Ample Time to Complete

Issue: For the case of two sequential frames, if a transfer descriptor from the first frame is delayed past the **PreSof** point of the second frame, then the transfer descriptor of the second frame is fetched even though it does not have ample time to be executed and completed.

Implication: Erroneous data and undefined USB behavior may result.

Workaround: None.

Status: Fixed in the B-0 stepping.

26. Mismatched PIDs Lead to False Bus Cycle

Issue: Mismatched PIDs may results under the following sequence:

1. The transfer is an Interrupt/Bulk/Control transfer.
2. The same end-point is being accessed.
3. The data buffer is not D-Word aligned (at an odd address).
4. The two data packets following each other have the same type (data0 followed by data0 or data1 followed by data1).

Implication: A false bus cycle is generated by the USB controller on the PCI bus to the data buffer with all byte enables inactive ("1111") and often with invalid data (in the case of an IN transfer).

Workaround: The following recommendations should be followed:

1. Do not locate data buffers at non d-word aligned addresses.
2. During this type of transfer always follow a data0 packet by a data1 packet and so on (i.e. do not cause a data toggle error).

Status: Fixed in the B-0 stepping.

27. Native Modem Bus Master Base Address

Issue: When bit 7 of the Native Modem Bus Mastering Base Address Register is set to 1, accesses to the Native Modem Bus Master Control Registers followed by accesses to the primary codec will only go to the secondary codec.

Implication: Since the primary codec cannot be accessed it can appear non-functional and the system modem could stop.

Workaround: The system BIOS must ensure that bit 7 of the Native Modem Bus Mastering Base Address Register is set to 0.

Status: Fixed in the B-0 stepping.

28. AC'97 PCM Out

Issue: The AC'97 controller in the 440MX will incorrectly insert one extra sample pair on the PCM Out channel under a specific condition. The sequence and conditions are described below.

- The last buffer in the current buffer list has been completely fetched and an odd number of buffers in this list has a length that is not a multiple of four.
- After the last buffer has been completely fetched, software updates the LVI.
- The first buffer in the next current buffer list has a length of exactly six samples.

When this occurs the 440MX will insert one extra sample pair, sent six frames (120 uS) earlier, between the last buffer of the old list and the first buffer of the new list.

Implication: The PCM Out channel will distort the intended sound. Since the time between these two buffers represents approximately 20 uS, the added sample should not be noticeable to the human ear.

Workaround: None.

Status: There are currently no plans to fix this erratum.

29. Incorrect Recognition of High/Low Speed USB Device Out of Resume

Issue: The USB Port Status and Control Register Low Speed Device Attached bit is incorrectly placed after a full-speed USB device generates K-state to resume system from POSCL, STR, and STD/SoftOff Suspend states. The bit will incorrectly identify the device as a low-speed device. This will only occur when the USB controller is going into suspend and Global Suspend is set before the Port Suspend is set.

Implication: Full-speed devices will be incorrectly identified as a low-speed device after resuming from POSCL, STR, and STD/SoftOff Suspend states.

Workaround: The workaround is separated into two time periods, before suspend and after resume.

Before Suspend: Ensure the BIOS sets the "Port Suspend" (bit 12 of USB IO register 10h/12h) before the "Enter Global Suspend Mode" (bit 3 of USB IO register 0) is set.

After Resume:

1. Add 20 ms or more delay before the BIOS clears the USB Command register bit 4 (FGR) after system resume from suspend.
2. Clear bit 4 Force Global Resume of USB Command register after resume.
3. Clear bit 3 Enter Global Suspend Mode of USB Command register after resume.
4. Clear bit 6 Resume Detect of both Port Status and Control register after resume.
5. Clear bit 12 of Suspend of both Port Status and Control registers on both ports after resume.

Status: There are currently no plans to fix this erratum.

30. Power Management General Purpose Status Register "USB_STS" Bit Not Cleared Correctly

Issue: The Power Management General Purpose Status Register USB_STS bit, bit 8, cannot be cleared after the system is resumed from POSCL, STR, and STD/SoftOff by K-state driven by a full speed USB device under the following conditions:

- When USB Command Register Enter Global Suspend Mode bit (bit 3) is set.
- USB Port Status and Control register Base+10-11h Suspend bit (bit 12) is not set.
- USB Command register Base+00-01h Enter Global Suspend Mode bit (bit 3) is not cleared.
- USB Port Status and Control register Base+10-11h Overcurrent Indicator bit (bit 11) is cleared.
- RSM_DET bit (bit 6) is cleared.
- Port Enable/Disable Change bit (bit 3) is cleared.
- Connect Status Change bit (bit 1) is cleared.

Implication: The 440MX will incorrectly identify that there is a potential wake up event on the USB when in an active state.

Workaround: The workaround is separated into two time periods, before suspend and after resume.

Before Suspend: Ensure the BIOS sets the "Port Suspend" (bit 12 of USB IO register 10h/12h) before the "Enter Global Suspend Mode" (bit 3 of USB IO register 0) is set.

After Resume:

1. Add 20 ms or more delay before the BIOS clears the USB Command register bit 4 (FGR) after system resume from suspend.
2. Clear bit 4 Force Global Resume of USB Command register after resume.
3. Clear bit 3 Enter Global Suspend Mode of USB Command register after resume.
4. Clear bit 6 Resume Detect of both Port Status and Control register after resume.
5. Clear bit 12 of Suspend of both Port Status and Control registers on both ports after resume.

Status: There are currently no plans to fix this erratum.

31. USB Port Status and Control Register "Connect Status Change" Bit Incorrectly Set

Issue: The USB Port Status and Control Register (PORTSC) Connect Status Change bit (bit 1) is not set to "1" when powered on.

Implication: The 440MX will not recognize if a device is connected to the USB port.

Workaround: When the system is first powered-on, perform host controller reset by setting USB Command Register bit #1.

Status: There are currently no plans to fix this erratum.

32. USB Rise/Fall Matching Errata

Issue: The Specification defines a rise/fall time matching (Trfm) which is calculated by dividing rise time by fall time (Tr/Tf). The specification for a full speed device is 90% minimum and 110% maximum. The 440MX does not meet this specification.

Implication: No failures have been reported in system validation to date as a result of this erratum. The 440MX does meet the required output signal crossover voltage specification (Vcrs).

Workaround: None

Status: There are currently no plans to fix this erratum.

33. USB Connect Status Change Bit does not get set under certain conditions

Issue: When system is in S0 state with CLKRUN# enabled and PCI bus is idle (and hence STP_PCI# is asserted), the USB Connect Status Change Bit does not get set.

Implication: The USB device will not be detected under the above condition. This may result in a no Plug and Play operation of the affected USB device

Workaround: There are four possible workarounds to solve this problem. The description of the workarounds is as follows:

1. Disable CLKRUN_EN: Clear CLKRUN_EN Register bit (<PM base>+10h bit 13 clear)
2. Using IO Trap: A BIOS workaround can be implemented using an IO trap mechanism to monitor activities on the USB host controller's port status and control registers.
3. Periodic Timer SMI: A BIOS workaround can be implemented using a Periodic Timer SMI to poll the activities on the USB host controller's port status and control registers.
4. USB device attach/de-attach detect: Hardware solution to detect when USB devices are attached. Allows enabling ACPI C3 when no USB devices are attached to the system.

Workaround Option #2: Register Setup for Workaround using IO Trap

When CLKRUN# is enabled (<PM base>+10h bit 13 set). Enable device IO trap (Assuming device 12 is used).

- Clear EIO_EN_DEV12 (dev7 func3 PCI 5Ch bit 29)
- Set IBASE_DEV12 (dev7 func3 PCI 68h bits 0-15) to USB base address+10h
- Set IMASK_DEV12 (dev7 func3 PCI 68h bits 16-19) to 3
- Set IO_EN_DEV12 (dev7 func3 PCI 68h bit 20)
- Set <PM base>+2Ch bit 24

Steps for Execution using IO Trap Workaround:

USB host controller port status and control registers (USB base + 10 - 13h) are periodically polled by OS/driver to check for status change. Accesses to these registers are trapped and SMI is generated.

- Confirm the SMI cause is from IO trap of these registers.
- Obtain the Opcode of IO instruction just executed based on the CS:EIP from the SMRAM save state map (translation may be required based on whether the system is running in real, VM86 or protected mode).

- Obtain the data value for the execution of the above IO instruction from the EAX slot of the SMRAM save state map.
- Determine if it is IO read or IO write instruction.
- If it is an IO read instruction, check Current Connect Status (USB base + 10h/12h bit 0).
- If no USB device is connected, no correction is needed.
- If an USB device is connected, check previous connect status flag saved in SMRAM.
- If no USB device was previously connected, it indicates a status change. Correction is needed.
- Patch the data value by setting the Connect Status Change (bit 1).
- Store data value back into the EAX slot of SMRAM save state map.
- Update the previous connect status flag.
- Exit SMI

Workaround Option #3:   Register Setup for Workaround using Periodic Timer

When CLKRUN# is enabled (<PM base>+10h bit 13 set). Enable Periodic Timer SMI. At each SMI, do the following:

- Check if CLKRUN_EN (<PM base>+10h bit 13) bit is set.
    - If NOT, then USB detection is not an issue. Exit SMI Handler routine and reset Timer.
    - If SET, then Read the USB Port Status and Control Register (USB Base + 10-13h).
        - Check for previously stored value of Connect Status bit.
        - Check Current Connect Status bit.
        - If a New Connection is detected (previous value of bit 1 is '0' and current value is '1'), then
            - Disable CLKRUN by clearing CLKRUN_EN bit.
            - Check if the device inserted is FULL or SLOW Speed Device.
            - If FULL SPEED:
                - Reset the Port
            - If SLOW SPEED:
                - Use GPIO Control to Reset the Vcc to the USB Port (Cycle the Power, ensuring the Vcc goes to 0 to properly Reset the Port).
            - Check if Connect Status (bit 1) is correctly set.
            - Store new value of Connect Status.
            - Enable CLKRUN by setting CLKRUN_EN bit.
        - EXIT SMI Handler

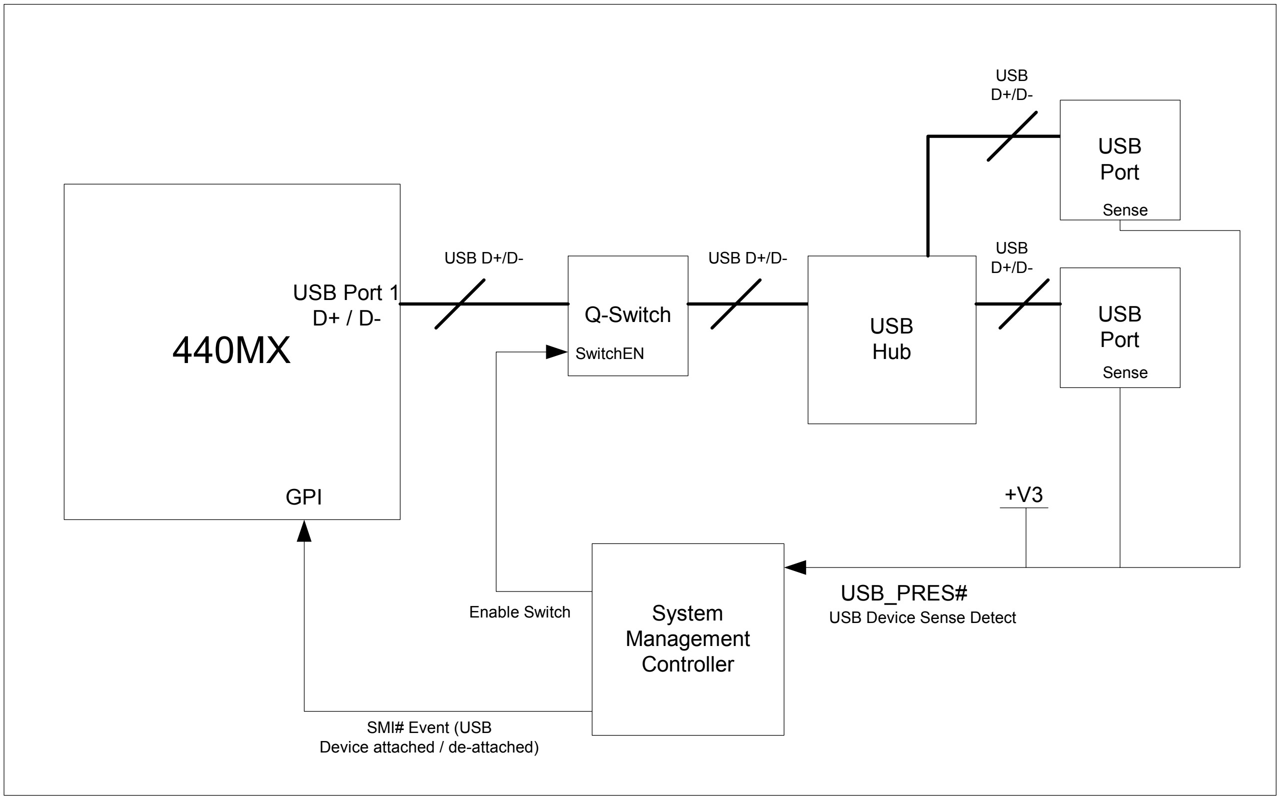
If the time to ensure proper USB Port Vcc cycling to 0 V is too long to be in a single SMI, the algorithm can be modified as follows:

- Disable CLKRUN by clearing CLKRUN_EN bit.
- Check if the device inserted is FULL or SLOW Speed Device.
- If FULL SPEED:
    - If 1st SMI, put Port into Reset by setting Port Reset bit.
        - Set Flag to mark 1st SMI completed.
    - If 2nd SMI, bring Port out of Reset
        - Check if Connect Status (bit 1) is correctly set.
        - Store new value of Connect Status.
- If SLOW SPEED:
    - If 1st SMI, Use GPIO Control to turn off the Vcc to the USB Port(s).
        - Set Flag to mark 1st SMI completed.
    - If 2nd SMI, Use GPIO Control to turn on the Vcc to the USB Port(s).
        - Check if Connect Status (bit 1) is correctly set.
        - Store new value of Connect Status.
- Enable CLKRUN by setting CLKRUN_EN bit.

Workaround Option #4: USB Device Attach/De-attach Hardware Detect

Circuitry can be implemented to allow the chipset to enter a power management state with an on-board USB hub. The 5th pin of each of a USB connector with "sense detect" is used to sense a USB device's presence (Note: not all USB ports have a 5th pin for sense detect, check part specification for details). The sense detect pins are wired together to form one signal USB_PRES# and connected to the System Management Controller (SMC). When there is no USB device connected, the 5th pin on the USB connector is floating and therefore the signal is pulled up to +V3_USB. When a USB device is plugged into any one of the USB connectors, the connector's 5th pin is grounded, pulling USB_PRES# low. When the USB_PRES# is pulled low, the SMC alerts the chipset via an SMI# and then enables the 2-bit Q-Switch which connects 440MX USB port to the on-board USB hub (see Figure 3 below). The Hub will identify itself to the chipset. When the chipset recognizes the hub, the hub informs the chipset of any devices connected to it (i.e., the device attached that caused USB_PRES# to be pulled low). When there are no devices connected to the USB connectors, the SMC disables the Q-Switch, which disconnects the on-board USB hub from the chipset and therefore allows the processor to enter power management states such as C3.

Figure 3: Errata 33, Workaround #4: USB Device Attach/De-attach - Hardware Solution



Status: There are currently no plans to fix this erratum.

34. AC97 Soft Audio and Soft Modem Issue during Intel® SpeedStep® Technology Operating Mode Transitions

Issue: Under certain stress test conditions, occasional system hang may occur during Intel® SpeedStep® Technology (which is featured in Intel mobile Pentium III processors) operating mode transition while running soft audio or soft modem applications. The AC97 DMA controller is not paused prior to Intel SpeedStep® technology operating mode transition and there could potentially be memory cycles between AC97 codec and DRAM as the system enters C3 State.

Implication: Audio and Modem may show abnormal behavior and the system may hang subsequently.

Workaround: The description of the workarounds is as follows. Please contact the Intel Field Sales Representative for the sample code of the BIOS workaround:

Note: Modem line drop could potentially occur even with the BIOS w/a. The recommendation is to disable the Intel SpeedStep® technology operating mode transition if active modem line is detected.

Pause the AC97 DMA controller prior to the Intel® SpeedStep® Technology operating mode transition to C3 State. The sequence of operation is as follows:

```
GeyservilleTransition PROC    FAR
```

A transition is needed and okay, save registers we will alter.
Now update the performance control field.
Turn off bus master reload.
Turn off bus master arbiter.
Check if AC97 modem is active. (Dev 0, Func 2, Offset 04h, Bit 0
Check if Ac97 modem is running (Read AC97 modem controller MBAR and check status of Run/Pause Bit)
Abort transition if modem is active. If not active,
Pause AC97 audio if it is running
Power down AC-link by doing a PR4 command to the codec.
Clear break event status.
Perform Level_3 read to enter C3 state.
Perform SpeedStep® transition sequence
Transition complete
Restore AC97 in reversed order!
Warm Reset AC97 codec
Restore Register values

Status: There are currently no plans to fix this erratum.

35. GPIO22/PIRQC# and GPIO23/PIRQD# Errata

Issue: When GPIO22 and GPIO 23 are selected (PIRQC# and PIRQD# are disabled), it has been observed that interrupts may not be appropriately serviced. It is recommended that GPIO22 and GPIO23 not be enabled.

Workaround: Do not select GPIO22 and GPIO23 and use other GPIO pins if needed.

Status: There are currently no plans to fix this erratum.

36. AC97 Soft Audio and Soft Modem Master Abort Errata

Issue: Use of either soft audio or soft modem on an Intel® 82443MX PCISet based platform running a 100-MHz Processor System Bus and an AC97 codec may result in failures. The system continues to function normally while the AC97 hardware may not resume and may require a cold-boot to recover. As a result of the failure, the Master Abort Status bit will be set in the audio or modem function PCI header space. (Bus 0, Device 0, Function 1(audio) or Function 2 (modem), offset 06-07h, bit 13).

Workaround: Intel has developed reference code for the workaround. Please contact the Intel Field Sales Representative for the sample code of the AC97 soft audio WDM driver workaround.

Note: The workaround needs to be enabled in different places in the driver based on whether it is a WavePCI or WaveCyclic soft audio driver implementation. Also, if it is a WDM soft modem driver, the placement will be different. Conceptually, the workaround remains the same.

The memory allocated by the driver for both the BDL and also the data buffers are marked un-cached. This eliminates write-backs to the soft audio memory locations and hence reduces the frequency of the failure significantly.

To set the memory un-cached after it has been allocated, do the following:

- Save the CR3 register of the processor
- Flush the Translation lookaside buffer (TLB) of the processor
- Detect the paging scheme
- If it is a 4MB paging scheme

- Get the base address from the CR3 of the processor
- Convert the physical base address to a virtual address
- Using the virtual address, traverse to the page directory and set the PCD bit. This will set the entire 4MB page un-cached.

- If it is a 4-KB paging scheme
  - Get the base address from the CR3 of the processor
  - Convert the physical base address to a virtual address
  - Using the virtual address, traverse to the page directory and get the physical base address for the page table entry.
  - Convert the physical base address to a virtual address.
  - Using the virtual address, traverse to the page table entry (PTE) and set the PCD bit. This will set the entire 4KB page un-cached.
- Restore the CR3 register of the processor
- Flush the Translation lookaside buffer (TLB) of the processor
- Invalidate caches and force a writeback (WBINVD).

Status: There are currently no plans to fix this erratum.

Specification Changes

There are no specification changes.

Specification Clarifications

There are no specification clarifications.

Documentation Changes

1. PME# Signal Description Correction

The PME# Signal Description is described incorrectly in Table 24 in *Intel® 82440MX PCIset Datasheet* which lists the Power on Reset values for various Signal Groups. PME# Signal is an Input to 440MX during POS, STR and STD and not Driven by 440MX as currently stated.

2. Thermal Design Power Change for 66-MHz Operation

The Thermal Design Power (TDP) for 440MX with 66-MHz Operating frequency is 1.7W. The previously stated TDP number, 1.6W, as stated in the *Intel® 82440MX PCIset Datasheet* is incorrect. The TDP for 100-MHz Operating frequency remains unchanged.

3. 440MX Processor Side Bus Vref Documentation Update

The GTL+ Reference Voltage description in Intel® 82443MX PCISet Electrical and Thermal Specification is updated for compatibility with LV/ULV Intel® Mobile Pentium® III processor-M AGTL Bus. Excerpt of **Table 1: 440MX DC Characteristics** of Intel® 82443MX PCISet Electrical and Thermal Specification (Datasheet Addendum, order #273502-001) has been provided below. Update shaded in gray.

Symbol	Parameter	Min	Max	Unit	Notes
GTL_REF	GTL+ Reference Voltage	5/9 V_{TT} - 2%	5/9 V_{TT} + 2%	V	For Intel® Mobile Pentium® II and Celeron™ (0.25u) designs
GTL_REF	GTL+ Reference Voltage	2/3 V_{TT} - 2%	2/3 V_{TT} + 2%	V	For Intel® Mobile Pentium® III and Celeron™ (0.18u) designs
AGTL_REF*	AGTL Reference Voltage	2/3 V_{TT} - 2%	2/3 V_{TT} + 2%	V	For LV/ULV Intel® Mobile Pentium® III processor-M (0.13u) designs

* References to Symbol GTL_REF found in Intel® 82443MX Datasheet and Intel® 82443MX Electrical and Thermal Specification (Datasheet Addendum) should be replaced with AGTL_REF voltages for LV/ULV Intel® Mobile Pentium® III processor-M (0.13u) based designs.

intel®

4. 440MX Processor Side Bus V_{TT} Documentation Update

The GTL+ Voltage Termination in the Intel® 82443MX PCISet Electrical and Thermal Specification has been updated for compatibility with LV/ULV Intel® Mobile Pentium® III processor-M AGTL bus. **Table 1: 440MX DC Characteristics** of Intel® 82443MX PCISet Electrical and Thermal Specification (Datasheet Addendum, order #273502-001) has been provided below. Update shaded in gray :

Symbol	Parameter	Min	Max	Unit	Notes
V_{TT}	GTL+ Termination Voltage (**VCCT = 1.5V**)	1.465	1.835	V	
V_{TT}	AGTL Termination Voltage (**VCCT = 1.25V**)	1.137	1.367	V	+/- 9% for platforms with P3P-M processor only

5. 440MX Processor Side Bus Undershoot/Overshoot Documentation Update

The Undershoot and Overshoot (VIL5 and VIH5) description in the Intel® 82443MX PCISet Electrical and Thermal Specification is updated for compatibility with LV/ULV Intel® Mobile Pentium® III processor-M AGTL Bus. An excerpt of **Table 1: 440MX DC Characteristics** of Intel® 82443MX PCISet Electrical and Thermal Specification (Datasheet Addendum, order #273502-001) has been provided below. The update is shaded in gray.

Symbol	Parameter	Min	Max	Unit	Notes
$V_{IL}5$	GTL+ Input Low Voltage	-0.3	GTL_REF – 0.2	V	1
$V_{IH}5$	GTL+ Input High Voltage	GTL_REF + 0.2	1.835	V	1
$V_{IL}5$	AGTL Input Low Voltage	-0.75	AGTL_REF – 0.2	V	2
$V_{IH}5$	AGTL Input High Voltage	AGTL_REF + 0.2	1.367	V	2

For Intel® Mobile Pentium® II processors and Mobile Celeron™ processors (0.25u) and Intel® Mobile Pentium® III processors and Mobile Celeron™ processor (0.18u) based designs.

References to symbol GTL_REF found in Intel® 82443MX Datasheet and Intel® 82443MX Electrical and Thermal Specification (Datasheet Addendum) should be replaced with AGTL_REF voltages for LV/ULV Intel® Mobile Pentium® III processor-M (0.13u) based designs.

6. Intel® 440MX PCIset NAND Tree Testing

This section provides information about the NAND tree testability features of the Intel 440MX PCI set.

The NAND tree A test mode is used by product engineers during manufacturing and OEMs during board level connectivity test. There are 8 separate NAND chains in NAND tree A and 1 chain in NAND tree B test mode. The NAND tree test modes can be entered as follows:

To enable the NAND tree test modes, it is **very important to first perform the Fast-Reset test mode** from cold start as shown in the waveform in section 1.1.2. Later the NAND tree test modes can be entered as shown in the waveform in section 1.1.3 i.e. the TEST# input pin is asserted low and a 4-bit value is presented on the PCI Requests i.e. PREQ[3:0]# input pins. The following table shows the PREQ[3:0]# signal encodings for the NAND tree test modes enabled via TEST# pin:

NAND Tree Test Modes

IRQ[4:3], PREQ[3:0]#	Test Mode Enabled	Comments
110000	NAND tree A	All 8 NAND Chains are enabled. They can be tested together or one by one each.
110001	NAND tree B	NAND-Outs of all 8 NAND tree-A chains connected.

Apart from the PREQ[3:0]# pins, IRQ[4:3] pins should be driven to 1 to enter the NAND tree test modes.

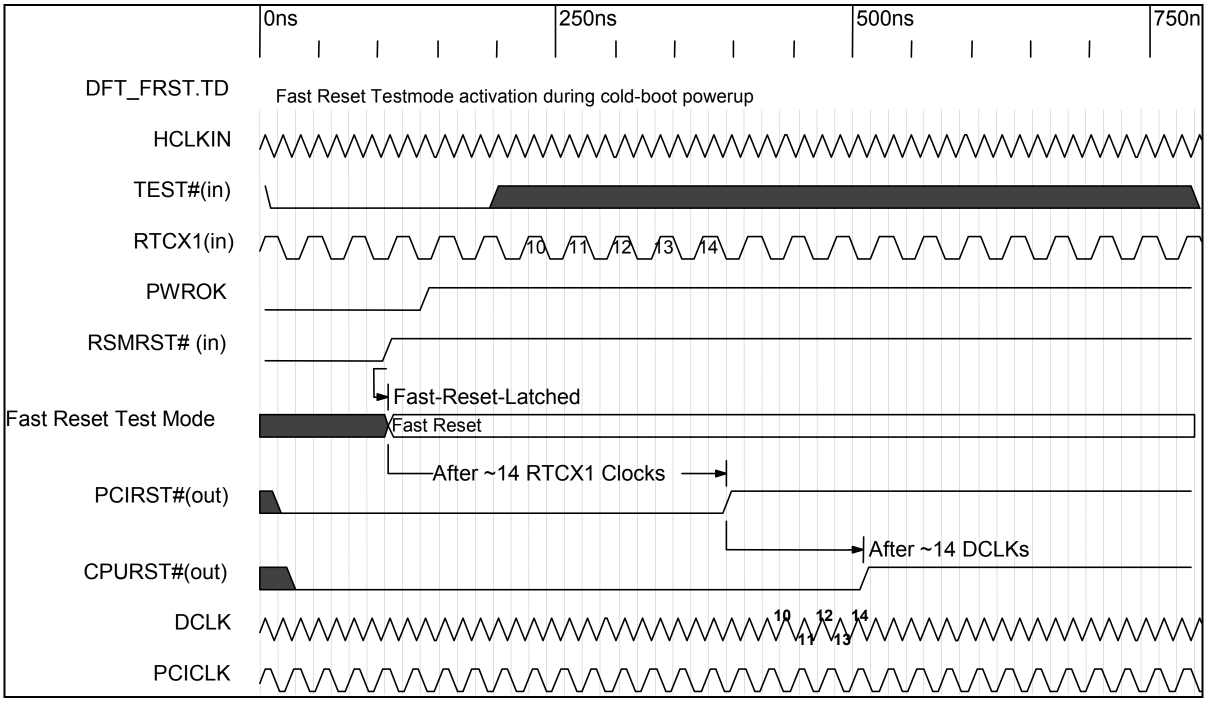
Fast Reset Test mode Activation

Actual system reset is approximately 1 ms to 4 ms long. To shorten system reset time for testing, a Fast Reset test mode has been implemented. In Fast Reset test mode, the CPURST# is de-asserted after ~14 DCLKs from PCIRST# sampled de-asserted.

The Fast Reset test mode is activated (i.e. Latched) if TEST# is driven '0' at the rising-edge of RSMRST#. In Fast Reset test mode, the CPURST# is only asserted for ~14 DCLKS after PCIRST# is sampled de-asserted. The following timing diagram illustrates these relationships

TEST#	RSMRST#	Test Mode Enable
0	↑(rising-edge)	Fast Reset Test mode enabled
1	↑(rising-edge)	Fast Reset Test mode disabled

Waveform of Fast Reset Test Mode



HCLKIN is not required to be running at all for the Fast Reset test mode to function.

Additional requirements for Fast Reset Test Mode:

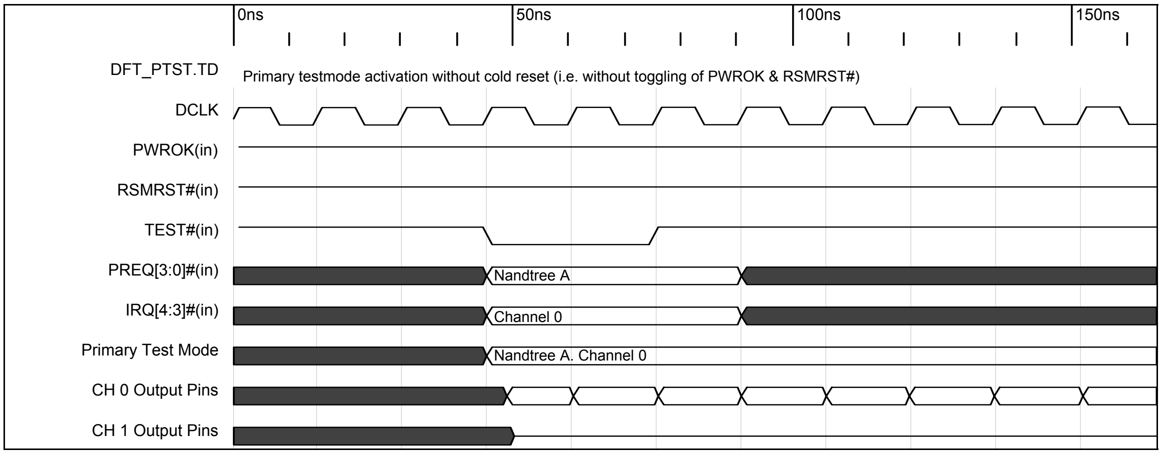
- The RTCX1 and PCICLK clocks are required to run from the beginning but the RTCX1 can be stopped after PCIRST# and CPURST# signals are sampled de-asserted
- The PCICLK "should" be stopped after PCIRST# and CPURST# signals are sampled de-asserted in order to check the continuity in NAND tree due to toggles on PCICLK pin.
- The only clock that should be running all the time is the DCLK.

Recommendations:

To enter the part into a NAND tree mode it is recommended that the appropriate values on PREQ[3:0]# be driven from the beginning in order to Latch-in the desired NAND tree test mode simultaneous to the execution of the Fast-Reset test mode. This avoids toggling of the TEST# pin multiple times.

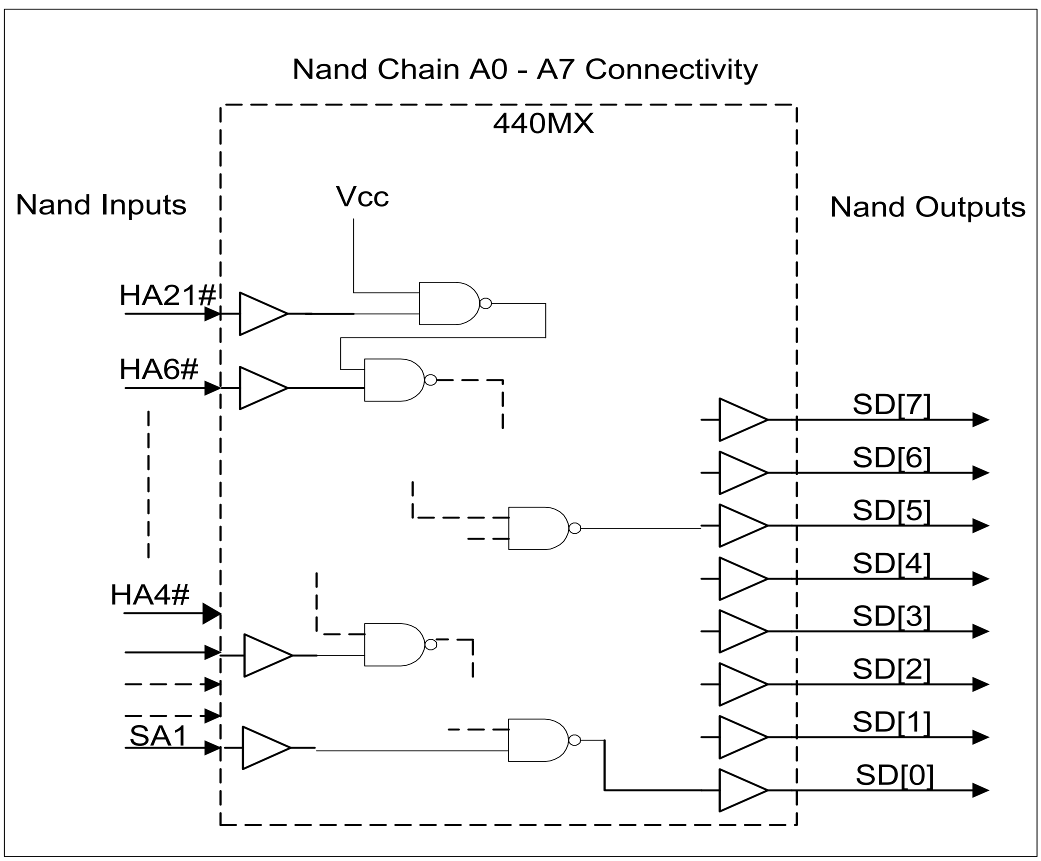
The following timing diagram shows the exact sequence required to enable NAND tree test mode. Note that the TEST# input pin acts as a latch enable, whereas the PREQ[3:0]# and IRQ[4:3]# pins act as latch inputs. The test mode is decoded from the output of the latch.

Waveform of NAND Tree Test modes



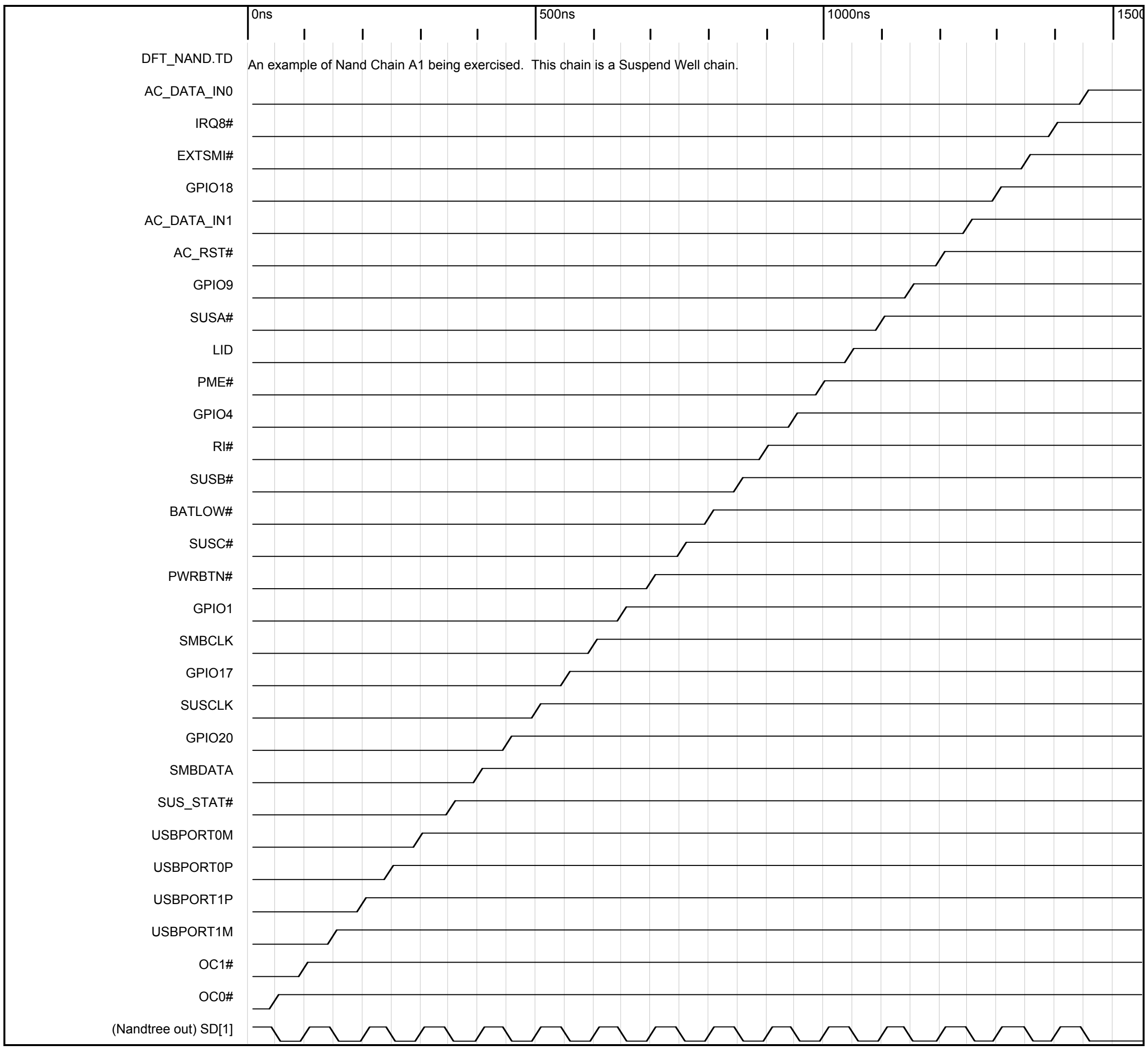
A conceptual diagram of the NAND tree is shown below.

Diagram of NAND Tree A Test mode (Chains A0-A7) Connectivity



An example NAND tree test waveform is shown below. At first, all the inputs pins are driven to logic 1. Next, each input pin is driven to logic 0, in a sequence, so that the output pin, in this case SD[7], toggles. By observing the NAND tree output pin, one can detect shorted and unconnected pins.

Waveform of NAND Chain A1 Test



NAND chain pin assignments are shown in the following tables.

Signals Not Included in NAND Tree A or NAND Tree B

Signals	Purpose
PWROK	Used for cold reset
RSMRST#	Used for cold reset
TEST#	Used to enter NAND tree A and B test modes

NAND Tree A Outputs

Signals	Purpose
SD[0]	NAND-Chain A0 Output
SD[1]	NAND-Chain A1 Output
SD[2]	NAND-Chain A2 Output
SD[3]	NAND-Chain A3 Output
SD[4]	NAND-Chain A4 Output
SD[5]	NAND-Chain A5 Output
SD[6]	NAND-Chain A6 Output
SD[7]	NAND-Chain A7 Output

Tables of NAND Tree A Chains

Note: Pins in NAND Chain #A1 reside in the suspend-well.

	Chain #A0		
#	Pin Name	Pad#	Ball
1	A20GATE	4	C4
2	IGNNE#	5	A2
3	RCIN#	6	D4
4	SERIRQ	7	F6
5	CPUSTP#	8	D3
6	SMI#	9	C2
7	INIT#	10	B2
8	PCISTP#	11	E4
9	IRQ14	12	F5
10	THRM#	13	H6
11	FERR#	14	B1
12	STPCLK#	15	E3
13	AC_SDATA_OUT	16	G5
14	AC_BIT_CLK	17	C1
15	AC_SYNC	18	D1
16	CLK48	64	N1
17	GPIO15	65	N5
18	SPKR	66	P1
19	ZEROWS#	68	P2
20	PCS0#	69	N4
21	PCS1#	70	P3
22	DACK3#	71	P4
23	IOCHRDY	78	R1
24	BIOSCS#	79	R4
25	RSTDRV	83	T4
26	MEMR#	84	U1
27	MEMW#	86	U2
28	DREQ2	87	U4
29	DREQ1	88	V1
30	SA18	89	U5
31	IOWB	90	U3
32	SA16	91	U6
33	SA9	94	W1
34	SA14	96	W2
35	SA15	98	Y1
36	SYSCLK	100	V3
37	IRQ6	102	Y2
38	IRQ5	103	W6
39	SA10	104	AA1
40	SA17	105	W4
41	SA4	106	AA2
42	SA12	107	W5
43	DACK2#	108	AB1
44	SA5	109	AB2
45	SA6	110	AA3
46	IOR#	111	Y4
47	SA0	112	AC1
48	SA11	114	AC2
49	DACK1#	115	Y5
50	SA2	116	AD1
48	SA8	117	AB3
49	IRQ1	118	AD2
50	IRQ3	119	Y3
51	SA13	120	AC3
52	DACK0#	121	AA4
53	SA1	122	AE1

Chain #A1		
Pin Name	Pad #	Ball
AC_SDATA_IN0	20	D2
IRQ8#	21	H5
EXSMI#	22	E2
GPIO18	24	G4
AC_SDATA_IN1	25	E1
AC_RST#	26	F2
GPIO9	27	J6
SUSA#	28	G3
LID	29	H4
PME#	30	F1
GPIO4	31	J5
RI#	33	J4
SUSB#	34	G2
BATLOW#	36	H3
SUSC#	37	G1
PWRBTN#	38	H2
GPIO1	39	F4
SMBCLK	40	J3
GPIO17	41	K4
SUSCLK	42	H1
GPIO20	43	K3
SMBDATA	44	J2
SUS_STAT#	45	L3
USBPRT0-	49	L4
USBPRT0+	50	J1
USBPRT1+	53	K2
USBPRT1-	54	L2
OC1#	55	M4
OC0#	56	L1

Chain #A2		
Pin Name	Pad #	Ball
GPIO13	72	R3
DREQ3	73	P5
KBCCS#	75	R5
MCCS#	76	T5
SA3	124	AE2
OSC	125	AB4
IRQ7	126	AE3
IRQ12	127	AF2
IRQ4	128	AF3
TC	130	AD4
SA7	131	AC4
DREQ0	132	AE4
PDCS3#	138	AC5
PDA1	139	AA6
PIORDY	140	AA5
PDD0	141	AA7
PDDAK#	142	AB5
PDCS1#	144	AB6
PDIOW#	145	AB7
GPIO5	146	AF4
PDD2	147	AD5
PDD15	148	AC6
PDD4	150	AE5
PDDRQ	151	AF5
PDD5	152	AC7
PDA2	153	AD7
PDD1	154	AE6
PDD12	155	AA9
PDIOR#	156	AF6
PDD10	158	AC8
PDD7	159	AA8
PDD11	160	AE7
PDD8	161	AB9
PDD3	162	AD8
PDD13	164	AF7
PDD9	165	AA10
PDD14	166	AE8
PDD6	167	AB8
PDA0	168	AF8
MD4	170	AD9
MD0	172	AE9
MD5	174	AC10
MD1	176	AF9
MD3	178	AE10
MD2	180	AF10
MD6	184	AE11
MD7	186	AF11
MD10	188	AE12
MD8	192	AF12
MD9	194	AF13
MD12	196	AE13
MD13	202	AF14

Chain #A3		
Pin Name	Pad #	Ball
MD32	173	AC9
MD33	177	AB10
MD37	182	AD11
MD35	183	AB11
MD38	190	AB12
MD34	191	AC11
MD36	195	AC12
MD39	198	AD13
MD42	199	AD12
MD41	200	AC13
MD46	203	AC15
MD43	206	AC14
MD44	207	AB15
MD40	208	AB14
MD45	217	AB16
MD47	226	AD17
MA5#	227	AB18
CS1#	228	AE17
MA0#	230	AE18
MA4#	231	AC18
CS0#	232	AF19
MA3#	234	AE19
MA8#	236	AF20
WE#	238	AA19
CKE2#	240	AB19
MA7#	241	AE20
MA12#	242	AF21
CS2#	244	AD19
MA6#	246	AD20
MA2#	247	AC19
MA9#	248	AE21
CKE3#	250	AC20
MA13	251	AF22
MA11#	252	AD21
MD50	254	AF23
MA10	255	AE22
MD48	258	AE23
MD51	263	AF24
MD49	264	AE24
MD53	265	AD23
MD55	276	AF25
DQM2	277	AD25
MD52	278	AE25
MD54	281	AE26
MD59	282	AB24
MD56	290	AB26
MD58	292	AA25
DCLKO	293	AC24
MD57	294	AA26
MD60	298	Y24
MD61	300	Y25
MD62	302	Y26

	Chain #A4		
#	Pin Name	Pad #	Ball
1	MD15	204	AE14
2	MD14	210	AF15
3	DQM4	212	AE15
4	CS3#	213	AC16
5	MD11	214	AD15
6	DQM1	216	AF16
7	SRAS#	218	AE16
8	DQM0	220	AD16
9	MA1#	221	AC17
10	DQM5	222	AF17
11	SCAS#	224	AF18
12	CKE1#	225	AB17
13	CKE0#	237	AA18
14	MD16	256	AB20
15	MD17	259	AC21
16	MD23	260	AC22
17	MD19	262	AB21
18	MD18	266	AB22
19	MD22	272	AC23
20	MD24	273	AB23
21	MD21	274	AA22
22	MD25	280	AA23
23	DQM3	284	AC25
24	DQM6	285	AD26
25	DCLK	286	AB25
26	DQM7	288	AC26
27	MD20	289	Y21
28	MD30	296	Y23
29	MD28	297	W21
30	MD26	301	Y22
31	MD29	304	W23
32	MD27	305	W22
33	MD31	306	W24
34	MD63	308	V22
35	HIT#	309	V23
36	RS0#	310	W26
37	RS2#	311	V21
38	BREQ0#	312	W25
39	DBSY#	313	U21
40	RS1#	314	V24
41	HD35#	410	A24
42	HD39#	415	D22
43	HD32#	418	E21
44	HD37#	421	D21
45	HD33#	422	B23
46	HD46#	424	E20
47	HD36#	425	B22
48	HD51#	426	A22
49	HD38#	427	F19
50	HD34#	428	C21
51	HD44#	430	D20
52	HD47#	431	F18
53	HD43#	432	B21
54	HD48#	433	E19
55	HD42#	434	A21

Chain #A5		
Pin Name	Pad #	Ball
HREQ3#	316	V25
DRDY#	317	U22
ADS#	318	V26
HREQ2#	319	T22
HLOCK#	320	U23
HREQ1#	322	T23
HREQ4#	324	U25
DEFER#	325	R22
HTRDY#	326	T24
HITM#	328	U26
HA6#	329	R25
HREQ0#	330	T26
HA9#	331	T25
HA4#	332	R23
BPRI#	334	R24
BNR#	335	R26
HA10#	336	P23
HA7#	337	P22
HA3#	338	P25
HA14#	340	P26
HA15#	341	N26
HA8#	342	N24
HA19#	343	N25
HA25#	344	M26
HA22#	346	M25
HA5#	347	N22
HA11#	348	M24
HA28#	349	N23
HA12#	350	M23
HA13#	352	M22
HA21#	353	L23
HA17#	354	L26
HA16#	355	L22
HA24#	356	L25
HA29#	358	L24
HA30#	360	K26
HA23#	361	K22
HA31#	362	K25
CPURST#	364	J26
HD6#	366	J25
HA20#	367	K23
HD4#	368	H26
HD9#	369	J22
HCLKIN	370	J24
HD15#	372	H25
HA18#	373	J21
HA26#	374	J23
HD23#	375	H22
HD1#	376	G26
HD10#	378	G25
HD8#	379	H21
HD2#	380	E26
HD14#	381	F26
HA27#	382	H23
HD5#	384	F25

Chain #A6		
Pin Name	Pad #	Ball
HD0#	385	G24
HD7#	386	E25
HD12#	388	G23
HD11#	390	F23
HD19#	391	D26
HD18#	392	D25
HD22#	393	G22
HD26#	394	C26
HD13#	396	C25
HD31#	397	B26
HD17#	398	F24
HD3#	399	D24
HD30#	400	E23
HD29#	407	A25
HD16#	408	B25
HD25#	409	E22
HD20#	412	D23
HD21#	414	C24
HD24#	416	B24
HD28#	419	A23
HD27#	420	C23
AD5	498	A9
TRDY#	499	F10
AD9	500	B9
AD16	502	C9
AD12	503	E9
AD8	504	A8
SERR#	505	D8
AD2	506	B8
AD10	508	C8
STOP#	509	F9
AD7	510	A7
DEVSEL#	511	E8
AD4	512	B7
C/BE0#	514	B6
AD6	515	A6
AD0	516	D7
PLOCK#	517	F8
AD3	518	E7
AD1	519	A5
PREQ3#	520	A4
PGNT3#	521	B5
PIRQC#	523	C6
PIRQD#	524	E6
REQA#	525	D6
GNTA#	526	C5
PIRQB#	528	E5
PIRQA#	529	D5
A20M#	530	B4
NMI	532	A3
INTR	534	B3

Chain #A7		
Pin Name	Pad #	Ball
HD49#	436	C20
HD45#	437	D19
HD41#	438	B20
HD59#	439	E18
HD53#	440	A20
HD57#	442	B19
HD61#	443	F17
HD58#	444	A19
HD55#	445	D18
HD52#	446	C18
HD60#	448	B18
HD54#	449	E17
HD62#	450	A18
HD56#	451	D17
HD40#	452	C17
HD63#	454	D16
HD50#	455	E16
CLKRUN#	456	B17
AD28	458	A17
PGNT2#	459	D14
PGNT0#	460	E15
PREQ2#	461	E14
PGNT1#	462	C16
AD29	464	B16
AD27	465	C13
PREQ0#	466	D15
C/BE3#	467	D13
AD25	468	A16
AD31	470	C15
AD22	471	E13
AD26	472	B15
AD21	473	C12
PCICLK	474	A15
AD20	476	B14
AD23	477	D12
PREQ1#	478	A14
PCIRST#	479	E12
AD30	480	B13
C/BE2#	482	A13
FRAME#	484	A12
AD24	485	C11
PAR	486	B12
AD17	487	D11
AD13	488	A11
AD15	490	B11
AD11	491	D10
AD14	492	A10
AD18	493	E10
C/BE1#	494	B10
AD19	496	C10
IRDY#	497	D9

NAND Tree B

The NAND tree B test mode is necessary to test the SD[7:0] pins, which are not tested by the NAND tree A test mode because they are outputs in NAND tree A test mode. In NAND tree B test mode, the SD[7:0] signals become inputs and the CPURST# pin becomes the output.

Table of NAND Tree B Outputs

Signals	Purpose
CPURST#	NAND-Chain B0 Output

Table of NAND Tree B, Chain# B0

	Chain #B0		
#	Pin Name	Pad #	Ball
1	SD3	74	R2
2	SD7	80	T1
3	SD6	81	T3
4	SD2	82	T2
5	SD1	92	V2
6	SD5	95	V4
7	SD0	97	V5
8	SD4	99	V6

Diagram of NAND Tree B Test mode