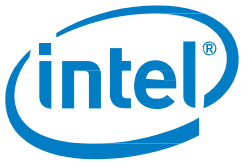


Intel® Atom™ Processor E6xx Series

Specification Update

July 2014

Revision 017

Document Number: 324209-017US

Contents

Revision History

Date	Revision	Description
September 2010	001	Initial release.
December 2010	002	• Added errata BI41 and BI44 and Specification Changes 1 though 4
February 2011	003	• Added Specification Change 2, and Document Changes 1 and 2
March 2011	004	• Added Specification Change 3, added section
April 2011	005	• Added Specification Changes 5, 7 and 8 • Corrected Table Number 4 and Specification Change 2 • Document Changes 3 through 6, 11, 14, and 15
May 2011	006	• Removed Specification Changes referring to Datasheet Rev. 002 and TMDG Rev. 001 • Added Specification Changes 9 and 10 • Document Changes 17 through 27, and 29
June 2011	007	• Added column “Stepping “to Table 5 through 7 • Added erratum BI45 and BI46 • Specification Changes 11 through 13 • Document only changes 30 and 32
August 2011	008	• Removed Specification Changes and Document Changes through Revision 007. • Added errata BI48, BI49, BI50. • Added Specification Change 1 • Specification Clarification 13, • Document-Only Changes 2 through 5.
November 2011	009	• Added erratum BI51 • Updated Table 10
February 2012	010	• Added Errata BI52, and BI53. • Specification Changes 2 through 4 • Documentation Changes 7 and 8. • Added B1 PRQ stepping to the Component Markings table.
April 2012	011	• Added Errata BI54 and BI55
May 2012	012	• Added Errata BI56
January 2013	013	• Added Errata BI57 and BI58
March 2013	014	• Added Errata BI59. Added Documentation Change 8
March 2013	015	• Out of cycle release to add Documentation Change 9.
May 2013	016	• Updated Erratum BI57.
July 2014	017	• Updated link to access *Intel® 64 and IA-32 Architectures Software Developer’s Manual*.

§

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.

Related Documents

Document Title	Document Number/Location
Intel® 64 and IA-32 Architectures Software Developer's Manuals	http://www.intel.com/content/www/us/en/processors/architectures-software-developer-manuals.html
Intel® Atom™ Processor E6xx *Series Datasheet*	http://download.intel.com/embedded/processor/datasheet/324208.pdf
Intel® Atom™ Processor E6xx *Series* Thermal *and Mechanical Design* Guidelines	http://download.intel.com/embedded/processors/thermalguide/ 324210.pdf

Nomenclature

Errata are design defects or errors. Errata may cause the Intel® Atom™ Processor E6xx Series' 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 Changes are modifications to the current published specifications. These changes will be incorporated in the next release of the specifications.

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 user documentation (datasheets, manuals, etc.).

§

Summary Tables of Changes

The following table indicates the Specification Changes, Errata, Specification Clarifications or Documentation Changes, which apply to the listed MCH 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

Stepping

X:	Erratum, Specification Change or Clarification that applies to this stepping.
(No mark) or (Blank Box):	This erratum is fixed in listed stepping or specification change does not apply to listed stepping.

Status

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

Row

Shaded:	This item is either new or modified from the previous version of the document.

Errata (Sheet 1 of 4)

Number	Steppings		Plans	ERRATA
BI1	X	X	No Fix	IO_SMI Indication in SMRAM State Save Area May Be Set Incorrectly
BI2	X	X	No Fix	Writes to IA32_DEBUGCTL MSR May Fail When FREEZE_LBRS_ON_PMI Set
BI3	X	X	No Fix	Address Reported by Machine-Check Architecture (MCA) on L2 Cache Errors May Be Incorrect
BI4	X	X	No Fix	Pending x87 FPU Exceptions (#MF) Following STI May Be Serviced before Higher Priority Interrupts
BI5	X	X	No Fix	Benign Exception After Double Fault May Not Cause Triple Fault Shutdown
BI6	X	X	No Fix	IA32_MC1_STATUS MSR Bit [60] Does Not Reflect Machine Check Error Reporting Enable Correctly
BI7	X	X	No Fix	If Two Logical Processors Use Same CR3 Value but Configure APIC Virtualization Differently, Either May Operate as if APIC Virtualization Were Disabled
BI8	X	X	No Fix	VM Exit Due to Fault While Delivering Software Interrupt May Save Incorrect Data Into VMCS
BI9	X	X	No Fix	VM Exit Occurring in IA-32e Mode May Not Produce VMX Abort When Expected
BI10	X	X	No Fix	Performance Monitoring Event for Outstanding Bus Requests Ignores AnyThread Bit
BI11	X	X	No Fix	Thermal Interrupts Dropped During and While Exiting Deep Power- Down State
BI12	X	X	No Fix	Corruption of CS Segment Register During RSM While Transitioning From Real Mode to Protected Mode
BI13	X	X	No Fix	Performance Monitoring Counter With AnyThread Bit Set May Not Count on Non-Active Thread
BI14	X	X	No Fix	GP and Fixed Performance Monitoring Counters With AnyThread Bit Set May Not Accurately Count Only OS or Only USR Events
BI15	X	X	No Fix	PMI Request Not Generated on Counter Overflow if Its OVF Bit Already Set in IA32_PERF_GLOBAL_STATUS
BI16	X	X	No Fix	Processor May Use Incorrect Translation if TLBs Contain Two Different Translations for Linear Address
BI17	X	X	No Fix	Write to APIC Register Sometimes May Appear to Have Not Occurred
BI18	X	X	No Fix	xTPR Update Transaction Cycle, if Enabled, May Be Issued to FSB After Processor Issued Stop-Grant Special Cycle
BI19	X	X	No Fix	Processor May Report #TS Instead of #GP Fault
BI20	X	X	No Fix	Writing Local Vector Table (LVT) When Interrupt Pending May Cause Unexpected Interrupt
BI21	X	X	No Fix	MOV To/From Debug Registers Causes Debug Exception
BI22	X	X	No Fix	Using 2M/4M Pages When A20M# Asserted May Result in Incorrect Address Translations
BI23	X	X	No Fix	Values for LBR/BTS/BTM Will Be Incorrect After Exit From SMM

Errata (Sheet 2 of 4)

Number	Steppings		Plans	ERRATA
BI24	X	X	No Fix	Incorrect Address Computed for Last Byte of FXSAVE/FXRSTOR Image Leads to Partial Memory Update
BI25	X	X	No Fix	Thermal Interrupt Not Generated When Current Temperature Invalid
BI26	X	X	No Fix	Programming Digital Thermal Sensor (DTS) Threshold May Cause Unexpected Thermal Interrupts
BI27	X	X	No Fix	Returning to Real Mode From SMM With EFLAGS.VM Set May Result in Unpredictable System Behavior
BI28	X	X	No Fix	Fault on ENTER Instruction May Result in Unexpected Values on Stack Frame
BI29	X	X	No Fix	With TF (Trap Flag) Asserted, FP Instruction That Triggers Unmasked FP Exception May Take Single Step Trap Before Retirement of Instruction
BI30	X	X	No Fix	Enabled Debug Breakpoint or Single Step Trap May Be Taken After MOV SS/POP SS Instruction if Followed by Floating Point Exception Signaling Instruction
BI31	X	X	No Fix	Code Segment Limit/Canonical Faults on RSM May Be Serviced Before Higher Priority Interrupts/Exceptions and May Push Wrong Address Onto Stack
BI32	X	X	No Fix	BTS (Branch Trace Store) and PEBS (Precise Event Based Sampling) May Update Memory Outside BTS/PEBS Buffer
BI33	X	X	No Fix	Single Step Interrupts With Floating Point Exception Pending May Be Mishandled
BI34	X	X	No Fix	Unsynchronized Cross-Modifying Code Operations Can Cause Unexpected Instruction Execution Results
BI35	X	X	No Fix	LBR Stack May Not Be Frozen on PMI Request When FREEZE_LBRS_ON_PMI Set
BI36	X	X	No Fix	PMI Request Not Generated on Counter Overflow if Its OVF Bit Already Set in IA32_PERF_GLOBAL_STATUS
BI37	X	X	No Fix	Synchronous Reset of IA32_MPERF on IA32_APERF Overflow May Not Work
BI38	X		Fixed	Processor May Not Recognize Signal PWROK on Its Initial Assertion
BI39	X		Fixed	Image on SDVO Display Clipped When Multiple Display Planes and SDVO Port Enabled
BI40	X	X	No Fix	Large Amounts of LPC Bus Memory Reads and Writes May Temporarily Starve Other I/O Devices
BI41	X	X	No Fix	Delayed I/O Device Memory Transactions Can Cause System Hang
BI42	X		Fixed	Software Can Inadvertently Change SDVO Base Class Code Register
BI43	X		Fixed	Voltage Supplied to Internal RTC Logic Violates Design Specification
BI44	X		Fixed	Flickering May be Observed on Display while Running Intensive Graphics and Video Decoding Activities
BI45	X	X	No Fix	C6 Request May Cause a Machine Check if the Other Logical Processor is in C4 or C6

Errata (Sheet 3 of 4)

Number	Steppings		Plans	ERRATA
BI46	x	X	No Fix	EOI Transaction May Not be Sent if Software Enters Core C6 During an Interrupt Service Routine
BI47	X	X	No Fix	PCNT Throttling May Cause System Hang During TM1 Thermal Event
BI48	X	X	No Fix	VMX Transitions May Set Bits 63:32 of the IA32_FMASK MSR
BI49	X	X	No Fix	Writing the Local Vector Table (LVT) when an Interrupt is Pending May Cause an Unexpected Interrupt
BI50	X	X	No Fix	Clearing PCIe* Root Port's BME Bit With Pending Upstream Traffic Will Cause Internal Bus to Hang
BI51	X	X	No Fix	CPUID Instruction Returns Incorrect Brand String
BI52	X		No Fix	The APIC Timer May Drift When Bus Ratios Less Than 6 Are Used
BI53	X	X	No Fix	Extended Tags Are Always Used But Not Reported on PCIe* Root Ports
BI54	X	X	No Fix	Outbound MSI From The PMU Can Result in Live Lock And/or System Hang When Simultaneously Occurring With an Inbound I/O Read
BI55	x	X	No Fix	SMBus Timing Violation
BI56	X	X	No Fix	RTC Does Not Detect a Coin Cell Battery Low Voltage Condition
BI57	x	X	No Fix	Complex Conditions Associated With Instruction Page Remapping or Self/Cross-Modifying Code Execution May Lead to Unpredictable System Behavior
BI58			No Fix	REP MOVS/STOS Executing With Fast Strings Enabled and Crossing Page Boundaries with Inconsistent Memory Types May Use an Incorrect Data Size or Lead to Memory-Ordering Violations
BI59	x	x	No Fix	Paging Structure Entry May be Used Before Accessed And Dirty Flags Are Updated

Specification Changes

No.	Stepping		Document Title	Rev.	Specification Changes
	B0	B1			
1	X	X	*Intel® Atom™ Processor E6xx Series Thermal and Mechanical Design Guidelines*	003	Heatsink Height Change
2	X	X	*Intel® Atom™ Processor E6xx Series Datasheet*	004	Changed VCC33RTC Specification in "Operating Condition Power Supply and Reference DC Characteristics" Table
3	X		*Intel® Atom™ Processor E6xx Series Datasheet*	004	Changed VCCRTCEXT Specification from "Operating Condition Power Supply and Reference DC Characteristics" Table
4	X	X	*Intel® Atom™ Processor E6xx Series Datasheet*	004	Changed VIH and VIL Specifications for RTCRST#, PWROK, and RSMRST# in "Active Signal DC Characteristics" Table

Specification Clarifications

No.	Stepping		Document Title	Rev.	Specification Clarifications
	B0	B1			
1	X	X	*Intel® Atom™ Processor E6xx Series Datasheet*	003	Add Information Related to Memory-Mapped Accesses

Document Changes (Sheet 1 of 2)

No.	Stepping		Document Title	Rev.	Document-Only Changes
	B0	B1			
1.	X	X	Intel® Atom™ Processor E6xx Series Datasheet	003	
2.	X	X	Intel® Atom™ Processor E6xx Series Datasheet	003	Missing 06h (Host Data 0) Register Description for SMBus Controller
3.	X	X	Intel® Atom™ Processor E6xx Series Datasheet	003	Missing 07h (HoDescription for SMBus Controllerst Data1) Register
4.	X	X	Intel® Atom™ Processor E6xx Series Datasheet	003	Changing BAR1 to WDTBA for WatchDog Timer Base Address Variable.
5.	X	X	Intel® Atom™ Processor E6xx Series Datasheet	003	Added VIH, VIL and ILEAK Specification and Remove VOH and IOH Specification to CMOS1.05 Open Drain

Document Changes (Sheet 2 of 2)

No.	Stepping		Document Title	Rev.	Document-Only Changes
	B0	B1			
6.		X	Intel® Atom™ Processor E6xx Series Datasheet	003	Updated GVD.FD Register Bit 0 Description
7.	X	X	Intel® Atom™ Processor E6xx Series Datasheet	003	Updated Note Text in the Operating Condition Power Supply and Reference DC Characteristics Table
8.		X	Intel® Atom™ Processor E6xx Series Datasheet	003	Corrected Bus 0, Device 3 PCI Configuration Register Default
9.		X	Intel® Atom™ Processor E6xx Series Datasheet	003	Update to Specification Change 4. Updated Notes for “Operating Condition Power Supply and Reference DC Characteristics” Table

§

Identification Information

Component Identification Using Programming Interface

The Intel® Atom™ Processor E6xx Series stepping can be identified by the following register contents:

Part Number	Stepping	CPUID	Vendor ID[1]	Device ID[2]	Rev Number[3]
Intel® Atom™ Processor E620 0.6 GHz (Commercial Temperature)	B0	0020661h	8086	4115	01
Intel® Atom™ Processor E620T 0.6 GHz (Extended Temperature)	B0	0020661h	8086	4115	01
Intel® Atom™ Processor E640 1.0 GHz (Commercial Temperature)	B0	0020661h	8086	4114	01
Intel® Atom™ Processor E640T 1.0 GHz (Extended Temperature)	B0	0020661h	8086	4114	01
Intel® Atom™ Processor E660 1.3 GHz (Commercial Temperature)	B0	0020661h	8086	4114	01
Intel® Atom™ Processor E660T 1.3 GHz (Industrial Temperature)	B0	0020661h	8086	4114	01
Intel® Atom™ Processor E680 1.6 GHz (Commercial Temperature)	B0	0020661h	8086	4114	01
Intel® Atom™ Processor E680T 1.6 GHz (Extended Temperature)	B0	0020661h	8086	4114	01
Intel® Atom™ Processor E620 0.6 GHz (Commercial Temperature)	B1	0020661h	8086	4115	02
Intel® Atom™ Processor E620T 0.6 GHz (Extended Temperature)	B1	0020661h	8086	4115	02
Intel® Atom™ Processor E640 1.0 GHz (Commercial Temperature)	B1	0020661h	8086	4114	02
Intel® Atom™ Processor E640T 1.0 GHz (Extended Temperature)	B1	0020661h	8086	4114	02
Intel® Atom™ Processor E660 1.3 GHz (Commercial Temperature)	B1	0020661h	8086	4114	02
Intel® Atom™ Processor E660T 1.3 GHz (Industrial Temperature)	B1	0020661h	8086	4114	02
Intel® Atom™ Processor E680 1.6 GHz (Commercial Temperature)	B1	0020661h	8086	4114	02
Intel® Atom™ Processor E680T 1.6 GHz (Extended Temperature)	B1	0020661h	8086	4114	02

NOTES:

1. The Vendor ID corresponds to bits [15:0] of the VID - Vendor Identification Register located at Offset 00–01h in the PCI Bus 0 Device 0 Function 0 configuration space.
2. The Device ID corresponds to bits [15:0] of the DID - Device Identification Register located at Offset 02–03h in the PCI Bus 0 Device 0 Function 0 configuration space.
3. The Revision Number corresponds to bits [7:0] of the RID - Revision Identification Register located at Offset 08h in the PCI Bus 0 Device 31 Function 0 configuration space.

Component Marking Information

The Intel® Atom™ Processor E6xx Series can be identified by the following component markings:

Stepping	S-Spec	Top Marking	Notes
B0 PRQ	909841	SLH94	1.6 GHz (Commercial Temperature)
B0 PRQ	905523	SLH54	1.3 GHz (Commercial Temperature)
B0 PRQ	905528	SLH55	1.0 GHz (Commercial Temperature)
B0 PRQ	905531	SLH56	0.6 GHz (Commercial Temperature)
B0 PRQ	909839	SLH95	1.6 GHz (Extended Temperature)
B0 PRQ	905804	SLH5L	1.3 GHz (Extended Temperature)
B0 PRQ	905805	SLH5M	1.0 GHz (Extended Temperature)
B0 PRQ	905806	SLH5N	0.6 GHz (Extended Temperature)
B1 ES	913796	QP4S	0.6 GHz (Commercial Temperature)
B1 ES	913784	QP4T	1.0 GHz (Commercial Temperature)
B1 ES	913750	QP4U	1.3 GHz (Commercial Temperature)
B1 ES	914597	QP4V	1.6 GHz (Commercial Temperature)
B1 QS	915059	QP7P	0.6 GHz (Extended Temperature)
B1 QS	915058	QP7Q	1.0 GHz (Extended Temperature)
B1 QS	915057	QP7R	1.3 GHz (Extended Temperature)
B1 QS	915056	QP7S	1.6 GHz (Extended Temperature)
B1 PRQ	913706	SLJ35	1.6 GHz (Commercial Temperature)
B1 PRQ	913720	SLJ34	1.3 GHz (Commercial Temperature)
B1 PRQ	913761	SLJ33	1.0 GHz (Commercial Temperature)
B1 PRQ	913773	SLJ32	0.6 GHz (Commercial Temperature)
B1 PRQ	913549	SLJ39	1.6 GHz (Extended Temperature)
B1 PRQ	913571	SLJ38	1.3 GHz (Extended Temperature)
B1 PRQ	913650	SLJ37	1.0 GHz (Extended Temperature)
B1 PRQ	913695	SLJ36	0.6 GHz (Extended Temperature)

Legend for Figure 1	Mark Text	Notes
GRP1INE1	INTEL {M} {C}'09 {e1}	1, 2, 3
GRP2LINE1	{FPO} {SPEC Code}	4
Legend for Figure 2	**Mark Text**	**Notes**
GRP1INE1	{FPO} {SPEC Code}	4
GRP2LINE1	INTEL {M} {C}'09 {e1}	1, 2, 3

NOTES:

1. M = Manufacturing copyright
2. NOTES:C = Copyright line
3. e1 = ROHS marking
4. FPO = Wafer lot #

Figure 1 Top-Side Marking Example for Intel® Atom™ Processor E6xx Series B0 Stepping.

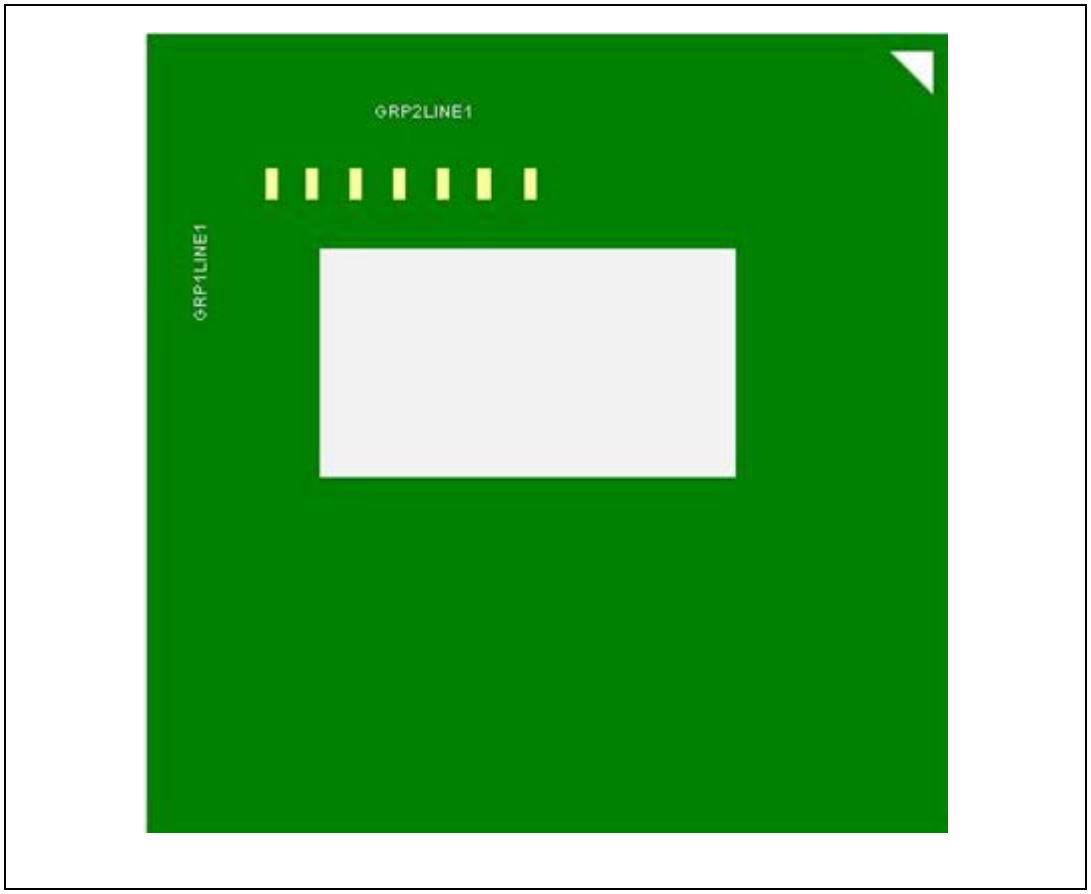
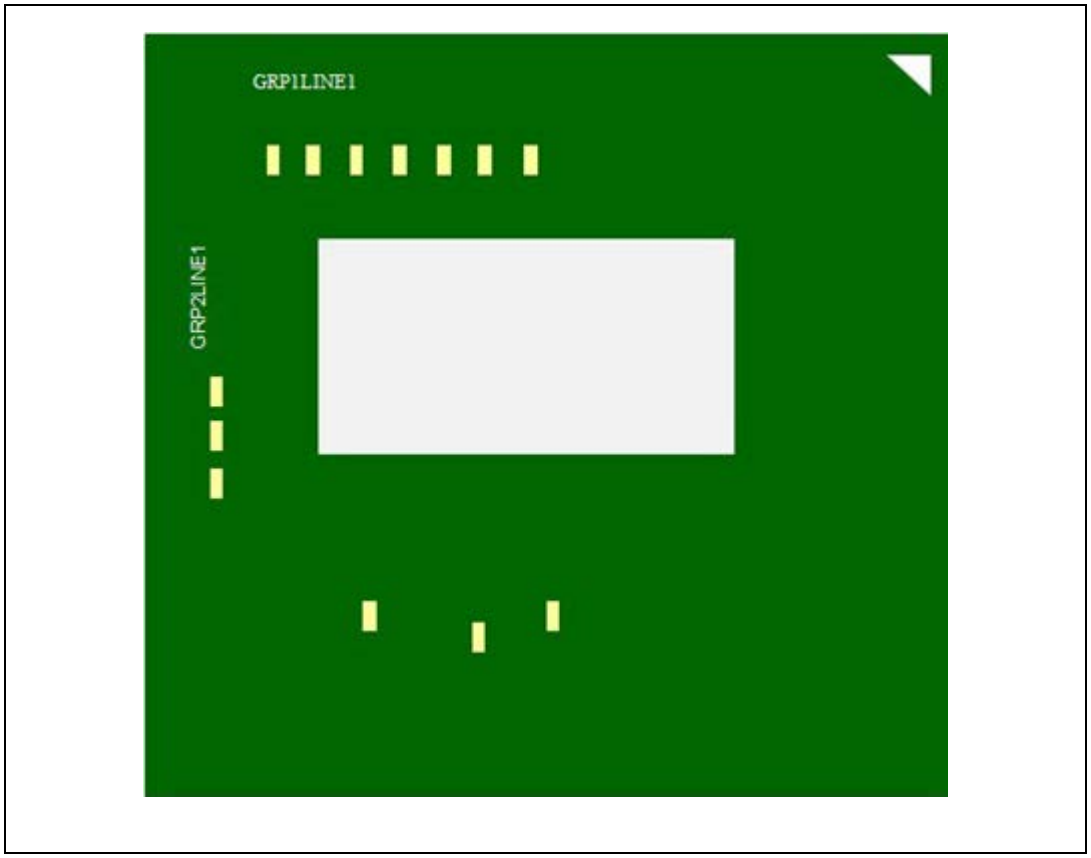


Figure 2. Top-Side Marking Example for Intel® Atom™ Processor E6xx Series B1 Stepping



Errata

BI1. IO_SMI Indication in SMRAM State Save Area May Be Set Incorrectly

Problem: The IO_SMI bit in SMRAM's location 7FA4h is set to "1" by the processor to indicate that a System Management Interrupt (SMI) occurred as the result of executing an instruction that reads from an I/O port. Due to this erratum, the IO_SMI bit may be incorrectly set by:

- An SMI that is pending while a lower priority event is executing
- A REP I/O read
- A I/O read that redirects to MWAIT

Implication: SMM handlers may get a false IO_SMI indication.

Workaround: The SMM handler has to evaluate the saved context to determine if the SMI was triggered by an instruction that read from an I/O port. The SMM handler must not restart an I/O instruction if the platform has not been configured to generate a synchronous SMI for the recorded I/O port address.

Status: No Fix.

BI2. Writes to IA32_DEBUGCTL MSR May Fail When FREEZE_LBRS_ON_PMI Set

Problem: When the FREEZE_LBRS_ON_PMI, IA32_DEBUGCTL MSR (1D9h) bit [11], is set, future writes to the IA32_DEBUGCTL MSR may not occur in certain rare corner cases. Writes to this register by software or during certain processor operations are affected.

Implication: Under certain circumstances, the IA32_DEBUGCTL MSR value may not be updated properly and will retain the old value. Intel has not observed this erratum with any commercially available software.

Workaround: Do not set the FREEZE_LBRS_ON_PMI bit of IA32_DEBUGCTL MSR.

Status: No Fix.

BI3. Address Reported by Machine-Check Architecture (MCA) on L2 Cache Errors May Be Incorrect

Problem: When an L2 Cache error occurs (error code 010Ah or 110Ah reported in IA32_MCi_STATUS MSR bits [15:0]), the address is logged in the MCA address register (IA32_MCi_ADDR MSR). Under some scenarios, the address reported may be incorrect.

Implication: Software should not rely on the value reported in IA32_MCi_ADDR MSR for L2 Cache errors.

Workaround: None identified.

Status: No Fix.

BI4. Pending x87 FPU Exceptions (#MF) Following STI May Be Serviced before Higher Priority Interrupts

Problem: Interrupts that are pending prior to the execution of the STI (Set Interrupt Flag) instruction are normally serviced immediately after the instruction following the STI. An exception to this is if the following instruction triggers a #MF. In this situation, the interrupt should be serviced before the #MF. Because of this erratum, if following STI, an instruction that triggers a #MF is executed while STPCLK#, Enhanced Intel SpeedStep® Technology transitions or Intel® Thermal Monitor events occur, the pending #MF may be serviced before higher priority interrupts.

Implication: Software may observe #MF being serviced before higher priority interrupts.

Workaround: None identified.

Status: No Fix.

BI5. Benign Exception After Double Fault May Not Cause Triple Fault Shutdown

Problem: According to the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3A, "Exception and Interrupt Reference," if another exception occurs while attempting to call the double fault handler, the processor enters shutdown mode. Due to this erratum, any benign faults while attempting to call the double fault handler will not cause a shutdown. However, Contributory Exceptions and Page Faults will continue to cause a triple fault shutdown.

Implication: If a benign exception occurs while attempting to call the double fault handler, the processor may hang or may handle the benign exception. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: No Fix.

BI6. IA32_MC1_STATUS MSR Bit [60] Does Not Reflect Machine Check Error Reporting Enable Correctly

Problem: IA32_MC1_STATUS MSR (405h) bit [60] (EN - Error Enabled) is supposed to indicate whether the enable bit in the IA32_MC1_CTL MSR (404h) was set at the time of the last update to the IA32_MC1_STATUS MSR. Due to this erratum, IA32_MC1_STATUS MSR bit [60] instead reports the current value of the IA32_MC1_CTL MSR enable bit.

Implication: IA32_MC1_STATUS MSR bit [60] may not reflect the correct state of the enable bit in the IA32_MC1_CTL MSR at the time of the last update.

Workaround: None identified.

Status: No Fix.

BI7. If Two Logical Processors Use Same CR3 Value but Configure APIC Virtualization Differently, Either May Operate as if APIC Virtualization Were Disabled

Problem: If a logical processor is in VMX non-root operation with the “virtual APIC accesses” VM-execution control set to 1, it may incorrectly operate as if the “virtual APIC accesses” VM-execution control was cleared to 0 if another logical processor has the same value in CR3 and one of the following is true:

- The other logical processor is not in VMX non-root operation.
- The other logical processor has the “virtual APIC accesses” VM-execution control cleared to 0.
- The other logical processor’s value of the “APIC-access address” VM-execution control field is different than that of the first logical processor.

Implication: A logical processor may fail to support the APIC-virtualization features properly if a virtual-machine monitor (VMM) uses the same page tables as a virtual machine (VM) using the APIC-virtualization features, or if two VMs (or two virtual CPUs within a VM) use the same page tables but operate with different settings of the APIC-virtualization features.

Workaround: A VMM should not use for itself the same page tables as a VM using the APIC-virtualization features, and it should configure two virtual CPUs to use the same page tables only if they use the same settings of the APIC-virtualization features.

Status: No Fix.

BI8. VM Exit Due to Fault While Delivering Software Interrupt May Save Incorrect Data Into VMCS

Problem: If a fault occurs during delivery of a software interrupt (INTn) in virtual-8086 mode when virtual mode extensions are in effect and that fault causes a VM exit, incorrect data may be saved into the VMCS. Specifically, information about the software interrupt may not be reported in the IDT-vectoring information field. In addition, the interruptibility-state field may indicate blocking by STI or by MOV SS if such blocking were in effect before execution of the INTn instruction or before execution of the VM-entry instruction that injected the software interrupt.

Implication: In general, VMM software that follows the guidelines given in the section “Handling VM Exits Due to Exceptions” of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B: System Programming Guide should not be affected. If the erratum improperly causes indication of blocking by STI or by MOV SS, the ability of a VMM to inject an interrupt may be delayed by one instruction.

Workaround: VMM software should follow the guidelines given in the section “Handling VM Exits Due to Exceptions” of the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B: System Programming Guide.

Status: No Fix.

BI9. VM Exit Occurring in IA-32e Mode May Not Produce VMX Abort When Expected

Problem: If a VM exit occurs while the processor is in IA-32e mode and the “host address-space size” VM-exit control is 0, a VMX abort should occur. Due to this erratum, the expected VMX aborts may not occur and instead the VM Exit will occur normally. The conditions required to observe this erratum are a VM entry that returns from SMM with the “IA-32e guest” VM-entry control set to 1 in the SMM VMCS and the “host address-space size” VM-exit control cleared to 0 in the executive VMCS.

Implication: A VM exit will occur when a VMX abort was expected.

Workaround: An SMM VMM should always set the “IA-32e guest” VM-entry control in the SMM VMCS to be the value that was in the LMA bit (IA32_EFER.LMA.LMA[bit 10]) in the IA32_EFER MSR (C0000080h) at the time of the last SMM VM exit. If this guideline is followed, that value will be 1 only if the “host address-space size” VM-exit control is 1 in the executive VMCS.

Status: No Fix.

BI10. Performance Monitoring Event for Outstanding Bus Requests Ignores AnyThread Bit

Problem: The Performance Monitoring Event of Outstanding Bus Requests will ignore the AnyThread bit (IA32_PERFEVTSEL0 MSR (186h)/IA32_PERFEVTSEL1 MSR (187h) bit [21]) and will instead always count all transactions across all logical processors, even when AnyThread is clear.

Implication: The performance monitor count may be incorrect when counting only the current logical processor's outstanding bus requests on a processor supporting Intel® Hyper-Threading Technology.

Workaround: None identified.

Status: No Fix.

BI11. Thermal Interrupts Dropped During and While Exiting Deep Power-Down State

Problem: Thermal interrupts are ignored while the processor is in the Deep Power-Down State as well as during a small window of time while exiting from the Deep Power-Down State. During this window, if the PROCHOT signal is driven or the internal value of the sensor reaches the programmed thermal trip point, the associated thermal interrupt may be lost.

Implication: In the event of a thermal event while a processor is waking up from the Deep Power-Down State, the processor will initiate an appropriate throttle response. However, the associated thermal interrupt generated may be lost.

Workaround: None identified.

Status: No Fix.

BI12. Corruption of CS Segment Register During RSM While Transitioning From Real Mode to Protected Mode

Problem: During the transition from real mode to protected mode, if an SMI (System Management Interrupt) occurs between the MOV to CR0 that sets PE (Protection Enable, bit 0) and the first far JMP, the subsequent RSM (Resume from System Management Mode) may cause the lower two bits of CS segment register to be corrupted.

Implication: The corruption of the bottom two bits of the CS segment register will have no impact unless software explicitly examines the CS segment register between enabling protected mode and the first far JMP. The Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3A: System Programming Guide, Part 1, in the section titled “Switching to Protected Mode,” recommends the far JMP immediately follows the write to CR0 to enable protected mode. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: No Fix.

BI13. Performance Monitoring Counter With AnyThread Bit Set May Not Count on Non-Active Thread

Problem: A performance counter with the AnyThread bit (IA32_PERFEVTSEL0 MSR (186h)/IA32_PERFEVTSEL1 MSR (187h) bit [21], IA32_FIXED_CTR_CTRL MSR (38Dh) bit [2] for IA32_FIXED_CTR0, bit [6] for IA32_FIXED_CTR1, bit [10] for IA32_FIXED_CTR2) set should count that event on all logical processors on that core. Due to this erratum, a performance counter on a logical processor which has requested to be placed in the Deep Power-Down State may not count events that occur on another logical processor.

Implication: The performance monitor count may be incorrect when the logical processor is asleep but still attempting to count another logical processor’s events. This will only occur on processors supporting Intel® Hyper-Threading Technology (Intel® HT Technology).

Workaround: None identified.

Status: No Fix.

BI14. GP and Fixed Performance Monitoring Counters With AnyThread Bit Set May Not Accurately Count Only OS or Only USR Events

Problem: A fixed or GP (general purpose) performance counter with the AnyThread bit (IA32_FIXED_CTR_CTRL MSR (38Dh) bit [2] for IA32_FIXED_CTR0, bit [6] for IA32_FIXED_CTR1, bit [10] for IA32_FIXED_CTR2; IA32_PERFEVTSEL0 MSR (186h)/IA32_PERFEVTSEL1 MSR (187h) bit [21]) set may not count correctly when counting only OS (ring 0) events or only USR (ring > 0) events. The counters will count correctly if they are counting both OS and USR events or if the AnyThread bit is clear.

Implication: A performance monitor counter may be incorrect when it is counting for all logical processors on that core and not counting at all privilege levels. This erratum will only occur on processors supporting multiple logical processors per core.

Workaround: None identified.

Status: No Fix.

BI15. PMI Request Not Generated on Counter Overflow if Its OVF Bit Already Set in IA32_PERF_GLOBAL_STATUS

Problem: If a performance counter overflows and software does not clear the corresponding OVF (overflow) bit in IA32_PERF_GLOBAL_STATUS MSR (38Eh), future overflows of that counter will not trigger PMI (Performance Monitoring Interrupt) requests.

Implication: If software does not clear the OVF bit corresponding to a performance counter, future counter overflows may not cause PMI requests.

Workaround: Software should clear the IA32_PERF_GLOBAL_STATUS.OVF bit in the PMI handler.

Status: No Fix.

BI16. Processor May Use Incorrect Translation if TLBs Contain Two Different Translations for Linear Address

Problem: The TLBs may contain both ordinary and large-page translations for a 4 kByte range of linear addresses. This may occur if software modifies a PDE (page-directory entry) that is marked present to set the PS bit (this changes the page size used for the address range). If the two translations differ with respect to page frame, permissions, or memory type, the processor may use a page frame, permissions, or memory type that corresponds to neither translation.

Implication: Due to this erratum, software may not function properly if it sets the PS flag in a PDE and also changes the page frame, permissions, or memory type for the linear addresses mapped through that PDE.

Workaround: Software can avoid this problem by ensuring that the TLBs never contain both ordinary and large-page translations for a linear address that differ with respect to page frame, permissions, or memory type.

Status: No Fix.

BI17. Write to APIC Register Sometimes May Appear to Have Not Occurred

Problem: With respect to the retirement of instructions, stores to the uncacheable memory based APIC register space are handled in a non-synchronized way. For example, if an instruction that masks the interrupt flag, for example, CLI, is executed soon after an uncacheable write to the Task Priority Register (TPR) that lowers the APIC priority, the interrupt masking operation may take effect before the actual priority has been lowered. This may cause interrupts whose priority is lower than the initial TPR, but higher than the final TPR, to not be serviced until the interrupt enabled flag is finally set, in other words, by an STI instruction. Interrupts will remain pending and are not lost.

Implication: In this example, the processor may allow interrupts to be accepted but may delay their service.

Workaround: This non-synchronization can be avoided by issuing an APIC register read after the APIC register write. This will force the store to the APIC register before any subsequent instructions are executed. No commercial operating system is known to be impacted by this erratum.

Status: No Fix.

BI18. xTPR Update Transaction Cycle, if Enabled, May Be Issued to FSB After Processor Issued Stop-Grant Special Cycle

Problem: According to the FSB (Front Side Bus) protocol specification, no FSB cycles should be issued by the processor once a Stop-Grant special cycle has been issued to the bus. If xTPR update transactions are enabled by clearing the IA32_MISC_ENABLES[bit 23] at the time of Stop-Clock assertion, an xTPR update transaction cycle may be issued to the FSB after the processor has issued a Stop Grant Acknowledge transaction.

Implication: When this erratum occurs in systems using C-states C2 (Stop-Grant State) and higher, the result could be a system hang.

Workaround: The BIOS must leave the xTPR update transactions disabled (default).

Status: No Fix.

BI19. Processor May Report #TS Instead of #GP Fault

Problem: A jump to a busy TSS (Task-State Segment) may cause a #TS (invalid TSS exception) instead of a #GP fault (general protection exception).

Implication: Operation systems that access a busy TSS may get an invalid TSS fault instead of a #GP fault. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: No Fix.

BI20. Writing Local Vector Table (LVT) When Interrupt Pending May Cause Unexpected Interrupt

Problem: If a local interrupt is pending when the LVT entry is written, an interrupt may be taken on the new interrupt vector even if the mask bit is set.

Implication: An interrupt may immediately be generated with the new vector when a LVT entry is written, even if the new LVT entry has the mask bit set. If there is no Interrupt Service Routine (ISR) set up for that vector, the system will GP fault. If the ISR does not do an End of Interrupt (EOI), the bit for the vector will be left set in the in-service register and mask all interrupts at the same or lower priority.

Workaround: Any vector programmed into an LVT entry must have an ISR associated with it, even if that vector was programmed as masked. This ISR routine must do an EOI to clear any unexpected interrupts that may occur. The ISR associated with the spurious vector does not generate an EOI; therefore, the spurious vector should not be used when writing the LVT.

Status: No Fix.

BI21. MOV To/From Debug Registers Causes Debug Exception

Problem: When in V86 mode, if a MOV instruction is executed to/from debug registers, a general- protection exception (#GP) should be generated. However, in the case when the general detect enable flag (GD) bit is set, the observed behavior is that a debug exception (#DB) is generated instead.

Implication: With debug-register protection enabled (in other words, the GD bit set), when attempting to execute a MOV on debug registers in V86 mode, a debug exception will be generated instead of the expected general-protection fault.

Workaround: In general, operating systems do not set the GD bit when they are in V86 mode. The GD bit is generally set and used by debuggers. The debug exception handler should check that the exception did not occur in V86 mode before continuing. If the exception did occur in V86 mode, the exception may be directed to the general-protection exception handler.

Status: No Fix.

BI22. Using 2M/4M Pages When A20M# Asserted May Result in Incorrect Address Translations

Problem: An external A20M# pin if enabled forces address bit 20 to be masked (forced to zero) to emulate real-address mode address wraparound at 1 megabyte. However, if all of the following conditions are met, address bit 20 may not be masked.

- Paging is enabled.
- A linear address has bit 20 set.
- The address references a large page.
- A20M# is enabled.

Implication: When A20M# is enabled and an address references a large page, the resulting translated physical address may be incorrect. This erratum has not been observed with any commercially available operating system.

Workaround: Operating systems should not allow A20M# to be enabled if the masking of address bit 20 could be applied to an address that references a large page. A20M# is normally only used with the first megabyte of memory.

Status: No Fix.

BI23. Values for LBR/BTS/BTM Will Be Incorrect After Exit From SMM

Problem: After a return from SMM (System Management Mode), the processor will incorrectly update the LBR (Last Branch Record) and the BTS (Branch Trace Store), hence, rendering their data invalid. The corresponding data if sent out as a BTM on the system bus will also be incorrect. Note: This issue would only occur when one of the three above mentioned debug support facilities is used.

Implication: The value of the LBR, BTS, and BTM immediately after an RSM operation should not be used.

Workaround: None identified.

Status: No Fix.

BI24. Incorrect Address Computed for Last Byte of FXSAVE/FXRSTOR Image Leads to Partial Memory Update

Problem: A partial memory state save of the 512-byte FXSAVE image or a partial memory state restore of the FXRSTOR image may occur if a memory address exceeds the 64 kB limit while the processor is operating in 16-bit mode or if a memory address exceeds the 4 GB limit while the processor is operating in 32-bit mode.

Implication: FXSAVE/FXRSTOR will incur a #GP fault due to the memory limit violation as expected, but the memory state may be only partially saved or restored.

Workaround: Software should avoid memory accesses that wrap around the respective 16-bit and 32-bit mode memory limits.

Status: No Fix.

BI25. Thermal Interrupt Not Generated When Current Temperature Invalid

Problem: When the DTS (Digital Thermal Sensor) crosses one of its programmed thresholds it generates an interrupt and logs the event (IA32_THERM_STATUS MSR (019Ch) bits [9,7]). Due to this erratum, if the DTS reaches an invalid temperature (as indicated IA32_THERM_STATUS MSR bit [31]) it does not generate an interrupt even if one of the programmed thresholds is crossed and the corresponding log bits become set.

Implication: When the temperature reaches an invalid temperature the processor does not generate a thermal interrupt even if a programmed threshold is crossed.

Workaround: None identified.

Status: No Fix.

BI26. Programming Digital Thermal Sensor (DTS) Threshold May Cause Unexpected Thermal Interrupts

Problem: Software can enable DTS thermal interrupts by programming the thermal threshold and setting the respective thermal interrupt enable bit. When programming the DTS value, the previous DTS threshold may be crossed. This will generate an unexpected thermal interrupt.

Implication: Software may observe an unexpected thermal interrupt occur after reprogramming the thermal threshold.

Workaround: In the ACPI/OS, implement a workaround by temporarily disabling the DTS threshold interrupt before updating the DTS threshold value.

Workaround: No Fix.

BI27. Returning to Real Mode From SMM With EFLAGS.VM Set May Result in Unpredictable System Behavior

Problem: Returning back from SMM mode into real mode while EFLAGS.VM is set in SMRAM may result in unpredictable system behavior.

Implication: If SMM software changes the values of EFLAGS.VM in SMRAM, it may result in unpredictable system behavior. Intel has not observed this behavior in commercially available software.

Workaround: SMM software should not change the value of EFLAGS.VM in SMRAM.

Status: No Fix.

BI28. Fault on ENTER Instruction May Result in Unexpected Values on Stack Frame

Problem: The ENTER instruction is used to create a procedure stack frame. Due to this erratum, if execution of the ENTER instruction results in a fault, the dynamic storage area of the resultant stack frame may contain unexpected values (in other words, residual stack data as a result of processing the fault).

Implication: Data in the created stack frame may be altered following a fault on the ENTER instruction. Please refer to “Procedure Calls For Block-Structured Languages” in the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 1: Basic Architecture for information on the usage of the ENTER instructions. This erratum is not expected to occur in ring 3. Faults are usually processed in ring 0, and stack switch occurs when transferring to ring 0. Intel has not observed this erratum on any commercially available software.

Workaround: None identified.

Status: No Fix.

BI29. With TF (Trap Flag) Asserted, FP Instruction That Triggers Unmasked FP Exception May Take Single Step Trap Before Retirement of Instruction

Problem: If an FP instruction generates an unmasked exception with the EFLAGS.TF=1, it is possible for external events to occur, including a transition to a lower power state. When resuming from the lower power state, it may be possible to take the single step trap before the execution of the original FP instruction completes.

Implication: A single step trap will be taken when not expected.

Workaround: None identified.

Status: No Fix.

BI30. Enabled Debug Breakpoint or Single Step Trap May Be Taken After MOV SS/POP SS Instruction if Followed by Floating Point Exception Signaling Instruction

Problem: A MOV SS/POP SS instruction should inhibit all interrupts including debug breakpoints until after execution of the following instruction. This is intended to allow the sequential execution of MOV SS/POP SS and MOV [r/e]SP, [r/e]BP instructions without having an invalid stack during interrupt handling. However, an enabled debug breakpoint or single step trap may be taken after MOV SS/POP SS if this instruction is followed by an instruction that signals a floating point exception rather than a MOV [r/e]SP, [r/e]BP instruction. This results in a debug exception being signaled on an unexpected instruction boundary since the MOV SS/POP SS and the following instruction should be executed automatically.

Implication: This can result in incorrect signaling of a debug exception and possibly a mismatched Stack Segment and Stack Pointer. If MOV SS/POP SS is not followed by a MOV [r/e]SP, [r/e]BP, there may be a mismatched Stack Segment and Stack Pointer on any exception. Intel has not observed this erratum with any commercially available software or system.

Workaround: As recommended in the Intel® 64 and IA-32 Architectures Software Developer's Manual, the use of MOV SS/POP SS in conjunction with MOV [r/e]SP, [r/e]BP will avoid the failure since the MOV [r/e]SP, [r/e]BP will not generate a floating point exception. Developers of debug tools should be aware of the potential incorrect debug event signaling created by this erratum.

Status: No Fix.

BI31. Code Segment Limit/Canonical Faults on RSM May Be Serviced Before Higher Priority Interrupts/Exceptions and May Push Wrong Address Onto Stack

Problem: Normally, when the processor encounters a Segment Limit or Canonical Fault due to code execution, a #GP (General Protection Exception) fault is generated after all higher priority Interrupts and exceptions are serviced. Due to this erratum, if RSM (Resume from System Management Mode) returns to the execution flow that results in a Code Segment Limit or Canonical Fault, the #GP fault may be serviced before a higher priority interrupt or exception (for example, NMI (Non-Maskable Interrupt), Debug break (#DB), Machine Check (#MC), etc.). If the RSM attempts to return to a non-canonical address, the address pushed onto the stack for this #GP fault may not match the non-canonical address that caused the fault.

Implication: Operating systems may observe a #GP fault being serviced before higher priority interrupts and exceptions. Intel has not observed this erratum on any commercially available software.

Workaround: None identified.

Status: No Fix.

BI32. BTS (Branch Trace Store) and PEBS (Precise Event Based Sampling) May Update Memory Outside BTS/PEBS Buffer

Problem: If the BTS/PEBS buffer is defined such that:

- The difference between the BTS/PEBS buffer base and the BTS/PEBS absolute maximum is not an integer multiple of the corresponding record sizes.
- The BTS/PEBS absolute maximum is less than a record size from the end of the virtual address space.
- The record that would cross the BTS/PEBS absolute maximum will also continue past the end of the virtual address space.

A BTS/PEBS record can be written that will wrap at the 4G boundary (legacy mode) or 2^{64} boundary (IA-32e mode, including both 64-bit mode and compatibility mode), and write memory outside of the BTS/PEBS buffer.

Implication: Software that uses BTS/PEBS near the 4G boundary (legacy mode) or 264 boundary (IA-32e mode, including both 64-bit mode and compatibility mode) and defines the buffer such that it does not hold an integer multiple of records can update memory outside the BTS/PEBS buffer.

Workaround: Define the BTS/PEBS buffer such that the BTS/PEBS absolute maximum minus the BTS/PEBS buffer base is an integer multiple of the corresponding record sizes as recommended in the Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B.

Status: No Fix.

BI33. Single Step Interrupts With Floating Point Exception Pending May Be Mishandled

Problem: In certain circumstances, when a floating point exception (#MF) is pending during single step execution, processing of the single step debug exception (#DB) may be mishandled.

Implication: When this erratum occurs, #DB will be incorrectly handled as follows:

- #DB is signaled before the pending higher priority #MF (interrupt 16).
- #DB is generated twice on the same instruction.

Workaround: None identified.

Status: No Fix.

BI34. Unsynchronized Cross-Modifying Code Operations Can Cause Unexpected Instruction Execution Results

Problem: The act of one processor, or system bus master, writing data into a currently executing code segment of a second processor with the intent of having the second processor execute that data as code is called cross-modifying code (XMC). XMC that does not force the second processor to execute a synchronizing instruction, prior to execution of the new code, is called unsynchronized XMC. Software using unsynchronized XMC to modify the instruction byte stream of a processor can see unexpected or unpredictable execution behavior from the processor that is executing the modified code.

Implication: In this case, the phrase “unexpected or unpredictable execution behavior” encompasses the generation of most of the exceptions listed in *the Intel® 64 and IA-32 Architectures Software Developer’s Manual Volume 3A*: *System Programming Guide*, including a General Protection Fault (#GP) or other unexpected behaviors.

Workaround: In order to avoid this erratum, programmers should use the XMC synchronization algorithm as detailed in the *Intel® 64 and IA-32 Architectures Software Developer’s Manual Volume 3A: System Programming Guide*, section “Handling Self- and Cross-Modifying Code.”

Status: No Fix.

BI35. LBR Stack May Not Be Frozen on PMI Request When FREEZE_LBRS_ON_PMI Set

Problem: When the FREEZE_LBRS_ON_PMI flag (IA32_DEBUGCTL MSR (1D9h) bit [11]) is set on an Intel® Atom™ Processor E6xx Series, a PMI (performance monitor interrupt) request should cause the LBR and TR flags (IA32_DEBUGCTL MSR (1D9h) bit [1:0]) to be cleared and the LBR (last branch record) stack to stop being updated by branches/interrupts/exceptions. Due to this erratum, the processor may clear the LBR and TR flags but not stop the LBR stack from being updated when FREEZE_LBRS_ON_PMI is set and a PMI request occurs.

Implication: Following a PMI request, the LBRs may continue to be updated by branches/interrupts/exceptions even when FREEZE_LBRS_ON_PMI is set. The LBRs may contain values recorded after the PMI request.

Workaround: None identified.

Status: No Fix.

BI36. PMI Request Not Generated on Counter Overflow if Its OVF Bit Already Set in IA32_PERF_GLOBAL_STATUS

Problem: If a performance counter overflows and software does not clear the corresponding OVF (overflow) bit in IA32_PERF_GLOBAL_STATUS MSR (38Eh), future overflows of that counter will not trigger PMI (Performance Monitoring Interrupt) requests.

Implication: If software does not clear the OVF bit corresponding to a performance counter, future counter overflows may not cause PMI requests.

Workaround: Software should clear the IA32_PERF_GLOBAL_STATUS.OVF bit in the PMI handler.

Status: No Fix.

BI37. Synchronous Reset of IA32_MPERF on IA32_APERF Overflow May Not Work

Problem: When either the IA32_MPERF or IA32_APERF MSR (E7h, E8h) increments to its maximum value of FFFF_FFFF_FFFF_FFFFh, both MSRs are supposed to synchronously reset to 0h on the next clock. Due to this erratum, IA32_MPERF may not be reset when IA32_APERF overflows. Instead, IA32_MPERF may continue to increment without being reset.

Implication: Due to this erratum, software cannot reply on the synchronous reset of the IA32_MPERF register. The typical usage of IA32_MPERF/IA32_APERF is to initialize them with a value of 0; in this case, the overflow of the counter would not happen for ten years.

Workaround: None identified.

Status: No Fix.

BI38. Processor May Not Recognize Signal PWROK on Its Initial Assertion

Problem: During any system power-on event in which power had been removed from the processor (including S3), the processor may fail to recognize the PWROK input signal during power-up. This missed observation is dependent upon the processor temperature and can cause the system not to boot.

Implication: The system may fail to boot. Each processor may be affected at different temperature ranges.

Workaround: An external hardware workaround has been identified. Please refer to Intel® Atom™ Processor E6xx Series-based – Platform Design Guide for details.

Status: Plan Fix.

BI39. Image on SDVO Display Clipped When Multiple Display Planes and SDVO Port Enabled

Problem: An internal timing generator and the SDVO (Serial Digital Video Output) timing generator may become out of synch with each other. This condition only manifests when any combination of multiple display planes and the SDVO port are concurrently enabled and can result in a clipped image.

Implication: A clipped image will be observed on the SDVO display.

Workaround: A software workaround has been identified and may be implemented as a workaround for this erratum.

Status: Plan Fix.

BI40. Large Amounts of LPC Bus Memory Reads and Writes May Temporarily Starve Other I/O Devices

Problem: During heavy memory read and memory write traffic to the LPC (Low Pin Count) Bus, downstream transactions to I/O devices may be blocked for periods of 3.2 µs.

Implication: I/O devices may suffer a performance impact (for example, audio glitches). None identified.

Status: No Fix.

BI41. Delayed I/O Device Memory Transactions Can Cause System Hang

Problem: Back-to-back uncacheable write transactions issued by the processor to downstream I/O devices can cause indeterminate delays to completions of upstream memory transactions from I/O devices.

Implication: Long bursts of downstream transactions may cause system hangs and/or audio glitches.

Workaround: A BIOS workaround to increase the priority of the affected transactions has been identified.

Status: No Fix.

BI42. Software Can Inadvertently Change SDVO Base Class Code Register

Problem: Software that programs the unused SDVO (Serial Digital Video Out) base address register at offset 1Ch (Bus 0; Device 3; Function 0) inadvertently modifies the SDVO base class code register (Bus 0; Device 3; Function 0; Offset 9h) to an incorrect value.

Implication: Software that relies on the base class code could fail to function properly or will report the SDVO device incorrectly.

Workaround: A BIOS workaround has been identified.

Status: Plan Fix.

BI43. Voltage Supplied to Internal RTC Logic Violates Design Specification

Problem: An internal voltage regulator for the RTC (Real Time Clock) logic produces an overvoltage condition and potential damage to the internal logic.

Implication: The processor is not guaranteed to meet a 10 year reliability lifetime. The system may fail to boot over a period of time or exhibit inaccurate RTC clock operation.

Workaround: A workaround has been identified. Please refer to the Intel® Atom™ Processor E6xx Series Erratum: “Voltage Supplied to Internal RTC (Real Time Clock) Logic Violates Design Specification” FAQ, document number 324702, for further details.

Status: Plan Fix.

BI44. Flickering May be Observed on Display while Running Intensive Graphics and Video Decoding Activities

Problem: During intensive graphics and video playback activities, and with large numbers of display read requests, the display may experience flickering.

Implication: Flickering may be observed on displays connected to the integrated display engine.

Workaround: A workaround has been identified in BIOS and Intel® Embedded Media Graphics Driver.

Status: Plan Fix.

BI45. C6 Request May Cause a Machine Check if the Other Logical Processor is in C4 or C6

Problem: A machine check may be generated if a logical processor requests the C6 C-state and the other logical processor is in either the C4 or C6 C-states.

Implication: This erratum may result in unexpected machine-check exceptions.

Workaround: It is possible for the BIOS to contain a workaround for this erratum.

Status: No Fix.

BI46. EOI Transaction May Not be Sent if Software Enters Core C6 During an Interrupt Service Routine

Problem: If core C6 is entered after the start of an interrupt service routine but before a write to the APIC EOI (End of Interrupt) register, and the core is woken up by an event other than a fixed interrupt source the core may drop the EOI transaction the next time APIC EOI register is written and further interrupts from the same or lower priority level will be blocked.

Implication: EOI transactions may be lost and interrupts may be blocked when core C6 is used during interrupt service routines.

Workaround: Software should check the ISR register and if any interrupts are in service only enter C1.

Status: No Fix.

BI47. PCNT Throttling May Cause System Hang During TM1 Thermal Event

Problem: The TM1 (Thermal Monitor 1) thermal event and PCNT (Processor Control) mechanism can each independently control processor throttling and should work at the same time without issues. Due to this erratum, if PCNT throttling is enabled during a TM1 thermal event, the two throttling mechanisms may conflict with each other and leading to a system hang.

Implication: When this erratum occurs, the system may hang.

Workaround: A BIOS workaround has been identified and may be implemented as a workaround for this erratum.

Status: No fix.

BI48. VMX Transitions May Set Bits 63:32 of the IA32_FMASK MSR

Problem: Bits 63:32 of the IA32_FMASK MSR (C0000084H) are reserved and attempts to set them should fail. Due to this erratum, loads of this MSR as part of a VMX transition (from either the VM-exit MSR-load area or the VM-entry MSR-load area) may set any of these bits without causing the transition to fail.

Implication: Subsequent reads of the IA32_FMASK MSR (e.g. by RDMSR) will return a non-zero value for bits 63:32. Intel has not observed this erratum with any commercially available software.

Workaround: Software should ensure that bits 63:32 are all 0 in any entry for the IA32_FMASK MSR in any VM-exit MSR-load area or VM-entry MSR-load area.

Status: No Fix

BI49. Writing the Local Vector Table (LVT) when an Interrupt is Pending May Cause an Unexpected Interrupt

Problem: If a local interrupt is pending when the LVT entry is written, an interrupt may be taken on the new interrupt vector even if the mask bit is set.

Implication: An interrupt may immediately be generated with the new vector when a LVT entry is written, even if the new LVT entry has the mask bit set. If there is no Interrupt Service Routine (ISR) set up for that vector the system will GP fault. If the ISR does not do an End of Interrupt (EOI) the bit for the vector will be left set in the in-service register and mask all interrupts at the same or lower priority.

Workaround: Any vector programmed into an LVT entry must have an ISR associated with it, even if that vector was programmed as masked. This ISR routine must do an EOI to clear any unexpected interrupts that may occur. The ISR associated with the spurious vector does not generate an EOI, therefore the spurious vector should not be used when writing the LVT.

Status: No Fix.

BI50. Clearing PCIe* Root Port's BME Bit With Pending Upstream Traffic Will Cause Internal Bus to Hang

Problem: When there are upstream transactions in any of the PCIe* root port's buffers and the BME (Bus Master Enable) bit of that root port is cleared to disable the port as bus master, the completion from the BME clear configuration cycle becomes blocked by the pending upstream transactions. Given these conditions, the processor will time out waiting for the blocked configuration cycle.

Implication: Due to this erratum an internal bus will hang leading to: IERR, NMI, Reset, etc.

Workaround: During OS runtime, BIOS and device drivers can only clear the root port or end point BMEs after the attached end point devices are put into idle state with no pending upstream transactions. BIOS and OS must make sure there are no upstream transactions before clearing the BME and before BIOS hands off the boot process to the OS.

Status: No Fix.

BI51. CPUID Instruction Returns Incorrect Brand String Problem:

Problem: When a CPUID instruction is executed with EAX = 80000002H, 80000003H and 80000004H on an Intel® Atom™ processor, the return value contains the brand string Genuine Intel (R) CPU when it should have Intel(R) Atom(TM) CPU with the corresponding SKU (e.g.,: E620 or E660).

Implication: When this erratum occurs, the processor will report the incorrect brand string.

Workaround: It is possible for BIOS to contain a workaround for this erratum.

Status: No Fix.

BI52. The APIC Timer May Drift When Bus Ratios Less Than 6 Are Used

Problem: The APIC Timer may not decrement correctly when using bus ratios lower than 6. This may cause limited amounts of drift.

Implication: Due to this erratum, in the worst case of a nominal bus frequency of 100 MHz, a DCR (Divide Configuration Register) of 128 and a bus ratio of 4, the counter can drift by up to ~60 µs per timer countdown.

Workaround: It is possible for the BIOS to contain a workaround for this erratum. This workaround will disable bus ratios less than 6.

Status: No fix for B0. Ultra Low Frequency Mode is de-featured in B1.

BI53. Extended Tags Are Always Used But Not Reported on PCIe* Root Ports

Problem: Although the device's PCIe* root ports do not report 8-bit Extended Tag support in the DCAP.EFTS register and don't allow the enabling of Extended Tags in the DCTL.ETFE register, they will still generate transactions with Extended Tags.

Implication: PCIe* end points that do not support extended tags as required by the PCIe* Base Specification may fail.

Workaround: None identified.

Status: No fix

BI54. Outbound MSI From The PMU Can Result in Live Lock And/or System Hang When Simultaneously Occurring With an Inbound I/O Read

Problem: The Power Management Unit (PMU) is capable of generating Message Signaled Interrupts (MSI) to the processor. In specific corner cases, an outbound MSI from the PMU may occur simultaneously with an inbound I/O read resulting in live lock and/or a system hang.

Implication: When this erratum occurs, the processor may live lock and/or result in a system hang. The PMU will not be able to support MSI for display power up and thermal trip events.

Workaround: PMU MSIs have been disabled in firmware. Software is required to not enable the MSIs for thermal events and Display power up.

Status: No Fix.

BI55. SMBus Timing Violation

Problem: Due to this erratum, there is a violation of the specified requirement for SMB Hold Time. The processor hold time from the falling edge of SMB Data to the falling edge of SMB CLK is 1/4 the programmed SMB CLK period. In the case of a 100 kHz bus, this becomes 10 µs / 4 = 2.5 µs, which violates the 4.0 µs specification.

Implication: Intel has not observed any functional failures with any commercially available systems.

Workaround: Customers have the option to program for slower SMB CLK.

Status: No Fix.

BI56. RTC Does Not Detect a Coin Cell Battery Low Voltage Condition

Problem: The RTC's (Real Time Clock) TRC.RTCIO bit in the GPE0BLK (Offset 34H; bit 2) is not set to 1 when the coin cell battery voltage is below the replacement threshold.

Implication: Software is unable to report when the RTC battery voltage is below the replacement threshold.

Workaround: None identified.

Status: No Fix.

BI57. Complex Conditions Associated With Instruction Page Remapping or Self/Cross-Modifying Code Execution May Lead to Unpredictable System Behavior

Problem: Under a complex set of internal conditions, instruction page remapping, or self/cross modifying code events may lead to unpredictable system behavior

Implication: Due to this Erratum, unpredictable system behavior may be observed.

Workaround: None identified.

Status: No Fix.

BI58. REP MOVS/STOS Executing With Fast Strings Enabled and Crossing Page Boundaries with Inconsistent Memory Types May Use an Incorrect Data Size or Lead to Memory-Ordering Violations

Problem: Under the conditions described in the Software Developers Manual section "Fast String Operation," the processor performs REP MOVS or REP STOS as fast strings. Due to this erratum, fast string REP MOVS/REP STOS instructions that cross page boundaries from WB/WC memory types to UC/WP/WT memory types, may start using an incorrect data size or may observe memory ordering violations.

Implication: Upon crossing the page boundary, the following may occur, dependent on the new page memory type:

- UC: The data size of each read and write may be different than the original data size.
- WP: The data size of each read and write may be different than the original data size and there may be a memory ordering violation.
- WT: There may be a memory ordering violation.

Workaround: Software should avoid crossing page boundaries from WB or WC memory type to UC, WP or WT memory type within a single REP MOVS or REP STOS instruction that will execute with fast strings enabled.

Status: No Fix.

BI59. Paging Structure Entry May be Used Before Accessed And Dirty Flags Are Updated

Problem: If software modifies a paging structure entry while the processor is using the entry for linear address translation, the processor may erroneously use the old value of the entry to form a translation in a TLB (or an entry in a paging structure cache) and then update the entry's new value to set the accessed flag or dirty flag. This will occur only if both the old and new values of the entry result in valid translations.

Implication: Incorrect behavior may occur with algorithms that atomically check that the accessed flag or the dirty flag of a paging structure entry is clear and modify other parts of that paging structure entry in a manner that results in a different valid translation.

Workaround: Affected algorithms must ensure that appropriate TLB invalidation is done before assuming that future accesses do not use translations based on the old value of the paging structure entry.

Status: No Fix.

§

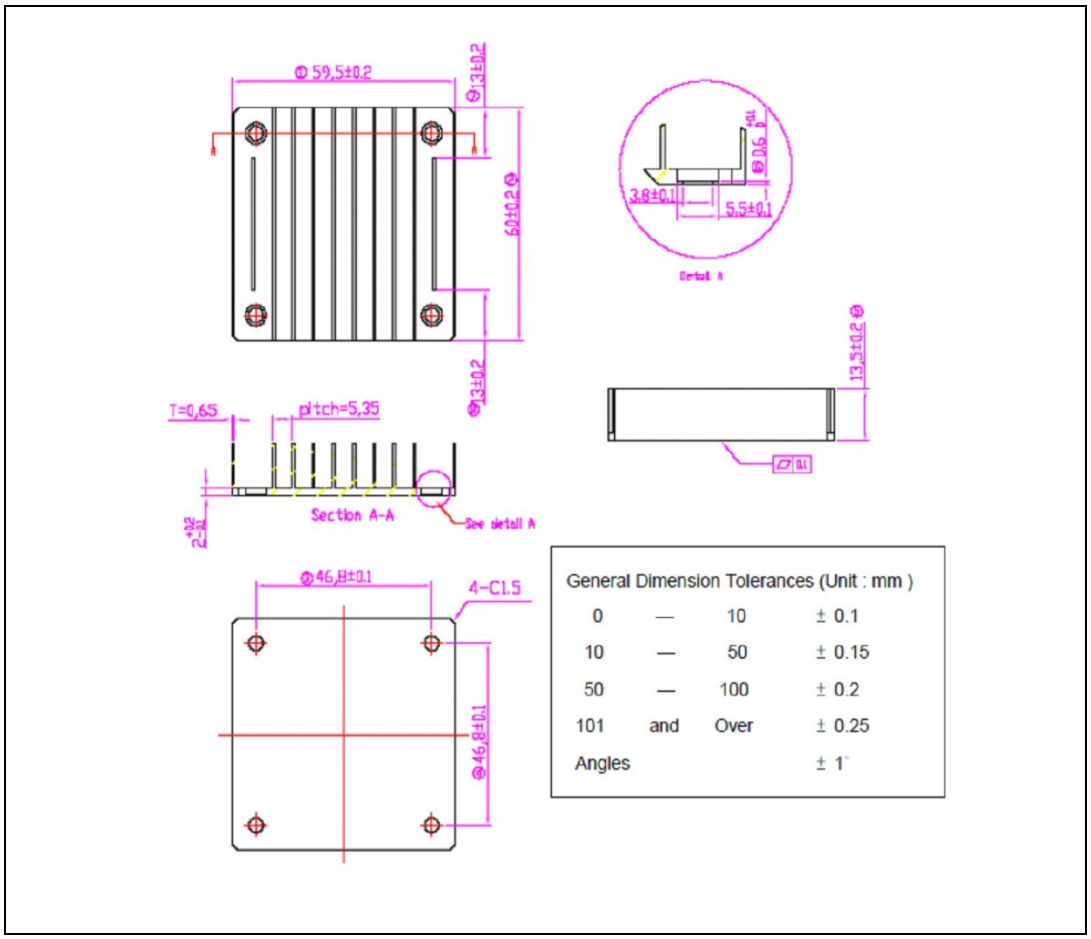
Specification Changes

1. Heatsink Height Change

Issue: Heatsink height on Figure 10 “1U Natural Convection Heatsink Mechanical Drawing” has changed from 13.5 ±0.2 mm to 29 ±0.2 mm.

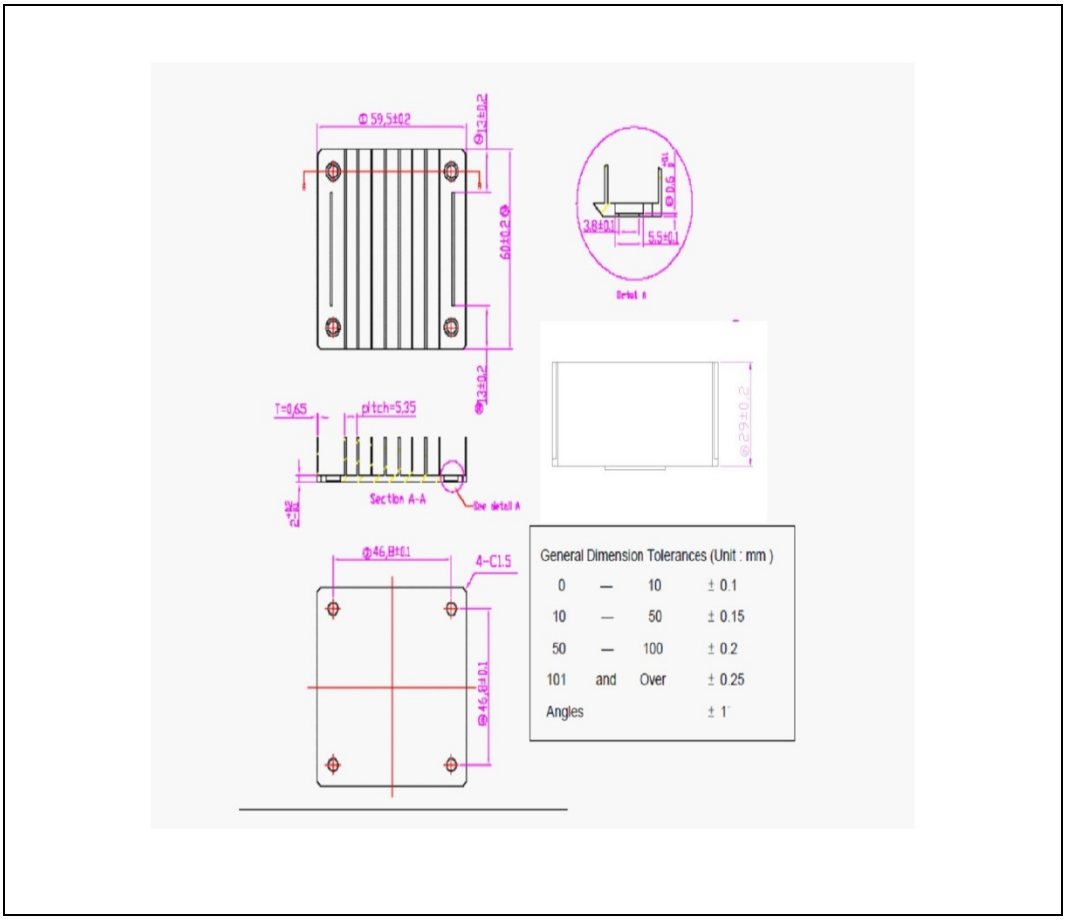
Old Text:

Figure 10. 1U Natural Convection Heatsink Mechanical Drawing



New Text:

Figure 10. 1U Natural Convection Heatsink Mechanical Drawing



Affected Docs: Figure 10 “1U Natural Convection Heatsink Mechanical Drawing” of *Intel® Atom™ Processor E6xx Series Thermal and Mechanical Design Guidelines*, Rev 003.

2. Changed VCC33RTC Specification in "Operating Condition Power Supply and Reference DC Characteristics" Table

Issue: As erratum of Voltage Supplied to Internal RTC Logic Violates Design Specification (Errata No. BI43) has been fixed in Intel® Atom™ Processor E6xx Series B1 stepping, the VCCRTCEXT specification is changed to reflect B0 and B1 stepping characteristics.

Old Text:

Table 405. Operating Condition Power Supply and Reference DC Characteristics [in Datasheet]

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
VCCP33, VCCP33SUS, VCCPSUS,	3.3 V Supply Voltage (Legacy IO, SDVO pads, Suspend Power supply, RTC suspend, RTC well)	3.135	3.3	3.465	V	
VCC33RTC	2.9 V Supply Voltage (RTC well)	2.0 (battery mode)	2.9	3.045	V	

New Text:

Table 405. Operating Condition Power Supply and Reference DC Characteristics [in Datasheet]

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
VCCP33, VCCP33SUS, VCCPSUS	3.3 V Supply Voltage (Legacy IO, SDVO pads, Suspend Power supply, RTC suspend)	3.135	3.3	3.465	V	
VCC33RTC	3.3 V Supply Voltage (RTC well)	3.135	3.3	3.465	V	B1 Stepping
		2.0 (battery mode)				
	2.9 V Supply Voltage (RTC well)	2.755	2.9	3.045	V	B0 Stepping
		2.17 (battery mode)				

Affected Docs: Table 405 "Operating Condition Power Supply and Reference DC Characteristics" of *Intel® Atom™ Processor E6xx Series Datasheet*, revision 004.

3. Changed VCCRTCEXT Specification from "Operating Condition Power Supply and Reference DC Characteristics" Table

Issue: As erratum of Voltage Supplied to Internal RTC Logic Violates Design Specification (Errata No. BI43) has been fixed in Intel® Atom™ Processor E6xx Series B1 stepping. The VCCRTCEXT specification is changed to apply only to the B0 stepping.

Old Text:

Table 405. Operating Condition Power Supply and Reference DC

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
VCCRTCEXT	1.24 V Supply Voltage (RTC well)	1.178	1.24	1.302	V	

New Text:

Table 405. Operating Condition Power Supply and Reference DC Characteristics

Symbol	Parameter	Min.	Typ.	Max.	Unit	Notes
VCCRTCEXT	1.24 V Supply Voltage (RTC well)	1.178	1.24	1.302	V	B0 Stepping only; does not apply to B1 stepping

Affected Docs: Table 405 "Operating Condition Power Supply and Reference DC Characteristics" of *Intel® Atom™ Processor E6xx Series Datasheet*, revision 004.

4. Changed VIH and VIL Specifications for RTCRST#, PWROK, and RSMRST# in "Active Signal DC Characteristics" Table

Issue: The VIH and VIL specifications are changed to reflect B0 and B1 stepping characteristics.

Old Text:

Table 406. Active Signal DC Characteristics

Symbol	Parameter	Min	Nom	Max	Unit	Notes
RTCRST#,PWROK,RSMRST#						
V_{IH}	Input high voltage	2.17		VCC33RTC + 0.5	V	
V_{IL}	Input low voltage	-0.5		0.68	V	

New Text:

Table 406. Active Signal DC Characteristics

Symbol	Parameter	Min	Nom	Max	Unit	Notes
RTCRST#,PWROK,RSMRST#						
V_{IH}	Input high voltage	2.0		VCC33RTC + 0.1	V	B1 Stepping
		2.17				B0 Stepping
V_{IL}	Input low voltage	-0.5		0.78	V	B1 Stepping
				0.68		B0 Stepping

Affected Docs: Table 406 "Active Signal DC Characteristics" of *Intel® Atom™ Processor E6xx Series Datasheet*, revision 004.

Specification Clarifications

1. Add Information Related to Memory-Mapped Accesses

Issue: New information is added to Section 5.2 “Introduction” related to memory-mapped accesses which cross DWORD boundary and lead to unpredictable processor behavior.

Old Text:

5.2 Introduction

The Intel® Atom™ Processor E6xx Series contains two sets of software accessible registers accessed via the host processor I/O address space: Control registers and internal configuration registers.

- Control registers are I/O mapped into the processor I/O space that controls access to PCI and PCI Express* configuration space.
- Internal configuration registers residing within the processor are partitioned into nine logical device register sets, one for each PCI device listed in Table 39. (These are “logical” devices because they reside within a single physical device.)

The processor’s internal registers (I/O Mapped, Configuration and PCI Express* Extended Configuration registers) are accessible by the host processor. The registers that reside within the lower 256 bytes of each device can be accessed as Byte, Word (16-bit), or DWord (32-bit) quantities, with the exception of CONFIG_ADDRESS, which can only be accessed as a DWord. All multi-byte numeric fields use little-endian ordering (i.e., lower addresses contain the least significant parts of the field). Registers which reside in bytes 256 through 4095 of each device may only be accessed using memory mapped transactions in DWord (32-bit) quantities. Some of the registers described in this section contain reserved bits. These bits are labeled Reserved. Software must deal correctly with fields that are reserved. On reads, software must use appropriate masks to extract the defined bits and not rely on reserved bits being any particular value. On writes, software must ensure that the values of reserved bit positions are preserved. That is, the values of reserved bit positions must first be read, merged with the new values for other bit positions and then written back.

Note: Software does not need to perform read, merge, and write operations for the configuration address register.

In addition to reserved bits within a register, the processor contains address locations in the configuration space of the Host Bridge entity that are marked either Reserved or Intel Reserved. The processor responds to accesses to reserved address locations by completing the host cycle. When a Reserved register location is read, a zero value is returned. (Reserved registers can be 8, 16, or 32 bits in size). Writes to Reserved registers have no effect on the processor. Registers that are marked as Intel Reserved must not be modified by system software. Writes to Intel Reserved registers may cause system failure. Reads from Intel Reserved registers may return a non-zero value.

Upon a Cold Reset, the processor sets all configuration registers to predetermined default states. Some default register values are determined by external strapping options. The default state represents the minimum functionality feature set required to successfully bring up the system; it does not represent the optimal system configuration. It is the responsibility of the system initialization software (usually BIOS) to properly determine the DRAM configurations, operating parameters and optional system features that are applicable configurations, operating parameters and optional system features that are applicable and to program the registers the registers the system memory accordingly.

New Text:

5.2 Introduction

The Intel® Atom™ Processor E6xx Series contains two sets of software accessible registers accessed via the host processor I/O address space: Control registers and internal configuration registers.

- Control registers are I/O mapped into the processor I/O space that controls access to PCI and PCI Express* configuration space.
- Internal configuration registers residing within the processor are partitioned into nine logical device register sets, one for each PCI device listed in Table 39. (These are “logical” devices because they reside within a single physical device.)

The processor’s internal registers (I/O Mapped, Configuration and PCI Express* Extended Configuration registers) are accessible by the host processor. The registers that reside within the lower 256 bytes of each device can be accessed as Byte, Word (16-bit), or DWord (32-bit) quantities, with the exception of CONFIG_ADDRESS, which can only be accessed as a DWord. All multi-byte numeric fields use little-endian ordering (i.e., lower addresses contain the least significant parts of the field). Registers which reside in bytes 256 through 4095 of each device may only be accessed using memory mapped transactions in DWord (32-bit) quantities. Some of the registers described in this section contain reserved bits. These bits are labeled Reserved. Software must deal correctly with fields that are reserved. On reads, software must use appropriate masks to extract the defined bits and not rely on reserved bits being any particular value. On writes, software must ensure that the values of reserved bit positions are preserved. That is, the values of reserved bit positions must first be read, merged with the new values for other bit positions and then written back.

Note: Software does not need to perform read, merge, and write operations for the configuration address register.

Software must not generate configuration requests from memory-mapped accesses that cross DWORD boundary, as this will result in unpredictable behavior.

In addition to reserved bits within a register, the processor contains address locations in the configuration space of the Host Bridge entity that are marked either Reserved or Intel Reserved. The processor responds to accesses to reserved address locations by completing the host cycle. When a Reserved register location is read, a zero value is returned. (Reserved registers can be 8, 16, or 32 bits in size). Writes to Reserved registers have no effect on the processor. Registers that are marked as Intel Reserved must not be modified by system software. Writes to Intel Reserved registers may cause system failure. Reads from Intel Reserved registers may return a non-zero value.

Upon a Cold Reset, the processor sets all configuration registers to predetermined default states. Some default register values are determined by external strapping options. The default state represents the minimum functionality feature set required to successfully bring up the system; it does not represent the optimal system configuration. It is the responsibility of the system initialization software (usually BIOS) to properly determine the DRAM configurations, operating parameters and optional system features that are applicable configurations, operating parameters and optional system features that are applicable and to program the registers the registers the system memory accordingly.

Affected Docs: Intel® Atom™ Processor E6xx Series Datasheet Rev003.

§

Documentation Changes

There are no documentation changes in this Specification Update revision.

§