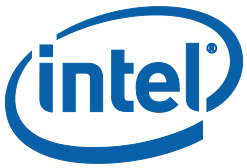


Intel® Atom™ Processor S1200 Product Family for Microserver

Specification Update

November 2012

Document Number: 328198-001

Revision History

Revision	Description	Date
1.0	Initial release.	November 2012

§

Contents

§

Preface

This document is an update to specifications contained in documents listed in the following Affected Documents/Related Documents table. It is a compilation of device and document errata, and specification clarifications and changes, intended for hardware system manufacturers and for software developers of applications, operating system, and tools. Information types defined in the Nomenclature are consolidated into this update document and are no longer published in other documents. This document may also contain information that has not been previously published.

Affected Documents

Document Title	Document Number/Location
Intel® Atom™ Processor S1200 Product Family for Microserver Datasheet Vol. 1 of 2	328194
Intel® Atom™ Processor S1200 Product Family for Microserver Datasheet Vol. 2 of 2	328195

NOTE:

1. Contact the local Intel representative for the latest revision and order number of this document.

Related Documents

Document Title	Document Number/Location
Intel® 64 and IA-32 Architecture Software Developer's Manual • Volume 1: Basic Architecture • Volume 2A: Instruction Set Reference Manual A-M • Volume 2B: Instruction Set Reference Manual N-Z • Volume 3A: System Programming Guide • Volume 3B: System Programming Guide *IA-32 Intel® Architecture Optimization Reference Manual*	325462
Intel® 64 and IA-32 Architecture Software Developer's Manual Documentation Changes	252046

NOTES:

1. Contact the local Intel representative for the latest revision and order number of this document.

Nomenclature

S-Spec Number is a five-digit code used to identify products. Products are differentiated by their unique characteristics, such as, core speed, L2 cache size, package type, etc. as described in the processor identification information table. Read all notes associated with each S-Spec number.

Errata: design defects or errors. Errata may cause the Intel® Atom™ Processor S1200 Product Family for Microserver 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: 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 current published documents. These changes will be incorporated in 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 Table of Changes

The following table indicates the errata, specification changes, specification clarifications or documentation changes that apply to the processor. 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.

Definitions are listed below for terminology used in the following Summary Tables.

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 the listed stepping or specification change does not apply to the listed stepping.

Status

Doc:	A 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 Number	B1	Plans	Errata
BY1	X	No Fix	Different INTx Deassertions in Close Succession May Cause SMBus Controller Misbehavior
BY2	X	No Fix	Spurious SMI May Occur During Warm Reset
BY3	X	No Fix	SMBus Controller May Send Spurious Legacy PCI Error Interrupts
BY4	X	No Fix	PCIe* Root Port May Transmit a Truncated DLLP
BY5	X	No Fix	The Processor May Hang Under Complex Conditions When Thermal Throttling Is Enabled
BY6	X	No Fix	Incorrect Logging of Masked PCIe Root Fabric AER Errors
BY7	X	No Fix	SMBus ERRUNCSEV Register Has The Wrong Default Value For Some Bits
BY8	X	No Fix	PCIe Header Log Register Contents Does Not Conform to Register Definition
BY9	X	No Fix	PCIe Root Port May Not Treat TLP as a Malformed TLP
BY10	X	No Fix	PCIe Ports May Log Receiver Overflow Error on an Unexpected Completion
BY11	X	No Fix	Thermal Trip May be Asserted During Cold Reset
BY12	X	No Fix	Writes to IA32_DEBUGCTL MSR May Fail when FREEZE_LBRS_ON_PMI is Set
BY13	X	No Fix	Synchronous Reset of IA32_MPERF on IA32_APERF Overflow May Not Work
BY14	X	No Fix	A Page Fault May Not be Generated When the PS bit is set to “1” in a PML4E or PDPTE
BY15	X	No Fix	LBR/BTM/BTS Information Immediately After a Transition From Legacy/Compatibility Mode to 64-Bit Mode May be Incorrect
BY16	X	No Fix	VID Information in IA32_PERF_STS MSR Bits [7:0] May be Incorrect
BY17	X	No Fix	GP and Fixed Performance Monitoring Counters With AnyThread Bit Set May Not Accurately Count Only OS or Only USR Events
BY18	X	No Fix	A VM Exit Occurring in IA-32e Mode May Not Produce a VMX Abort When Expected
BY19	X	No Fix	C6 Auto-Demotion Does Not Occur as Expected
BY20	X	No Fix	APIC Timer Can Expire Earlier Than Expected
BY21	X	No Fix	Core C-state Residency MSR Values Are Incorrect
BY22	X	No Fix	Local APIC Timer Expiration May Not be Honored if Software Requests Different Deep C-States For Each Logical Processor
BY23	X	No Fix	Task-Switching IRET May Not Guarantee That All Memory Writes Have Become Globally Visible
BY24	X	No Fix	An Indirect Jump Instruction May Execute to An Incorrect Target
BY25	X	No Fix	A VM Exit as a Result of a Task-Switching IRET May Not Properly Save the State of NMI Blocking
BY26	X	No Fix	Unsynchronized Cross-Modifying Code Operations Can Cause Unexpected Instruction Execution Results

Errata Number	B1	Plans	Errata
BY27	X	No Fix	Software Requested or Thermal Based Clock Modulation May Not Result in an Accurate Stop-Clock Duty Cycle
BY28	X	No Fix	Programming MSR_PKG_CST_CONFIG_CONTROL Package C-State Limit Field to 0x3 May Result in a System Hang
BY29	X	No Fix	CS Limit Violations May Not be Detected After VM Entry
BY30		No Fix	Fast-String Operations Are Not Enabled by Default
BY31		No Fix	PCIe* Root Ports May Incorrectly Indicate CRS Software Visibility Support

Number	Summary of Specification Changes
1	None to report at this time

Number	Summary of Specification Clarifications
1	None to report at this time

Number	Summary of Document Changes
1	None to report at this time

§

Identification Information

Component Identification via Programming Interface

The Intel® Atom™ Processor S1200 Product Family for Microserver (S12x0) stepping can be identified by the following register contents:

S12x0 Stepping	Features	Vendor ID	Device ID			Revision Number
			Intel® Atom™ Processor S1200 Product Family			
			S1260	S1240	S1220	
B1		0x8086	0x0C75	0x0C72	0x0C73	0x2

Component Marking Information

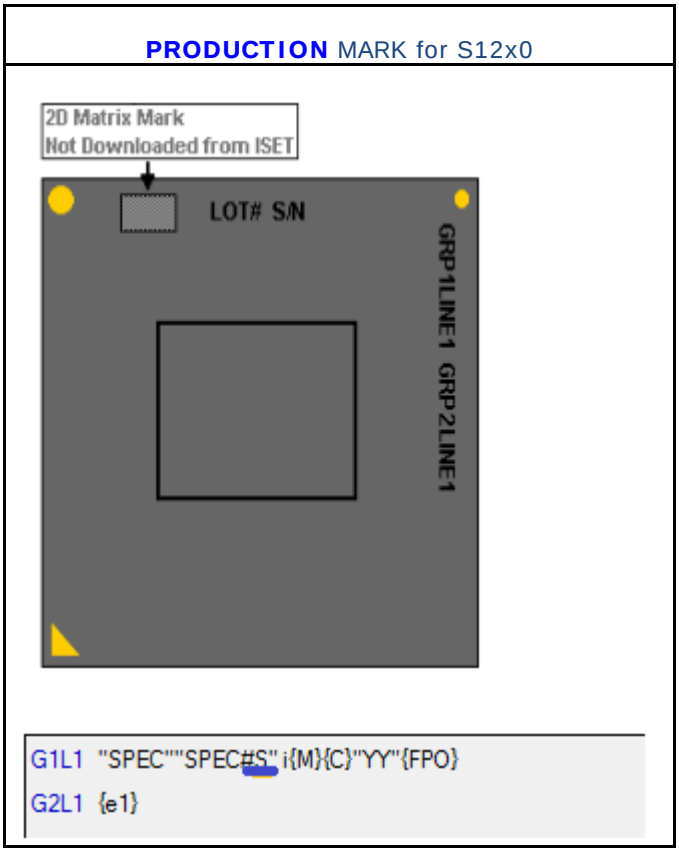
The Intel® Atom™ Processor S1200 Product Family for Microserver may be identified by the following component markings:

Stepping	SKU Name	S-Spec	MM	Notes
B1	Intel® Atom™ Processor S1260	SLK2H	925904	1
B1	Intel® Atom™ Processor S1240	SLK2J	925905	2
B1	Intel® Atom™ Processor S1220	SLK2K	925906	3

NOTES:

1. Speed: 2.0 GHz; Watt: 8.5W
2. Speed: 1.6 GHz; Watt: 6.3W
3. Speed: 1.6 GHz; Watt: 8.1W

Figure 1. Top Markings



§

Errata

BY1. Different INTx Deassertions in Close Succession May Cause SMBus Controller Misbehavior

Problem: Spurious interrupt deassertion glitches may happen in the SMBus controller if different INTx deassertions are closely spaced.

Implication: This erratum causes improper functioning of the SMBus controller.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY2. Spurious SMI May Occur During Warm Reset

Problem: Due to this erratum, during a warm reset a spurious SMI may be generated.

Implication: A spurious SMI during warm reset may cause the system to hang.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY3. SMBus Controller May Send Spurious Legacy PCI Error Interrupts

Problem: When operating in Legacy interrupt mode, the SMBus controller may not be able to suppress its legacy PCI error interrupt when the legacy PCI error interrupt is disabled.

Implication: Due to this erratum, the system may experience numerous unexpected interrupts.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY4. PCIe* Root Port May Transmit a Truncated DLLP

Problem: If a PCIe root port is configured to be an x2 link and it nullifies a transmitted TLP (Transaction Layer Packet), then due to this erratum, the root port may transmit a truncated DLLP (Data Link Layer Packet) subsequent to the nullified TLP.

Implication: When a link partner receives the truncated DLLP, it may signal a Framing Error and subsequently retrain the link.

Workaround: None identified.

Status: For the steppings affected, see the Summary Table of Changes.

BY5. The Processor May Hang Under Complex Conditions When Thermal Throttling Is Enabled

Problem: Processor may hang when a transaction is master aborted by the processor during thermal throttling state.

Implication: This erratum may cause the processor to hang when thermal throttling is enabled

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

Status: For the steppings affected, see the Summary Table of Changes.

BY6. Incorrect Logging of Masked PCIe Root Fabric AER Errors

Problem: When simultaneous multiple PCIe Root Fabric AER errors occur, the processor may log the AER Header of the higher priority error even if it is masked.

Implication: This erratum may impact the error logging capabilities of the processor.

Workaround: None identified.

Status: For the steppings affected, see the Summary Table of Changes.

BY7. SMBus ERRUNCSEV Register Has The Wrong Default Value For Some Bits

Problem: Due to this erratum, the uncorrectable error Severity Register for the SMBus controller (Bus 0; Device 19; Function 0, 1; offset 108H, bit 12, 15, 16, 20) is incorrectly initialized at reset.

Implication: By default, uncorrectable errors incurred by the SMBus controller may be signaled as fatal errors.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY8. PCIe* Header Log Register Contents Does Not Conform to Specification

Problem: The PCIe Header Log Register (Device 14; Function 0; Offset 1CH) captures the TLP (Transaction Layer Packet) header when an error is detected. Due to this erratum, the 32-bit items comprising the TLP header appear in reversed order with respect to the PCIe Base Specification definition for this register. That is, ordered as HDRLOG4..HDRLOG1 rather than the specified HDRLOG1..HDRLOG4.

Implication: Software attempting to access the Header Log Register may not behave as expected.

Workaround: None identified. It is possible to modify software accessing the Header Log Register to compensate for this erratum.

Status: For the steppings affected, see the Summary Table of Changes.

BY9. PCIe Root Port May Not Treat TLP as a Malformed TLP

Problem: A PCIe root port that receives a TLP (Transaction Layer Packet) with a data size greater than Max_Packet_Size must treat that TLP as a Malformed TLP. Due to this erratum, the processor may not treat the TLP as a Malformed TLP.

Implication: Receipt of TLPs with data sizes greater than Max_Packet_Size may lead to unpredictable system behavior. Intel has not observed this erratum with any commercially available system.

Workaround: None identified.

Status: For the steppings affected, see the Summary Table of Changes.

BY10. PCIe Ports May Log Receiver Overflow Error on an Unexpected Completion

Problem: Upon receipt of an Unexpected Completion, the PCIe Root Port (Bus 0; Device 1-4; Function 0) may log a Receiver Overflow error in addition to an unexpected completion error.

Implication: Due to this erratum, a Receiver Overflow error may be logged incorrectly. If Receiver Overflow errors are configured to be Fatal errors, this may result in a system hang.

Workaround: To avoid a hang, it is possible to configure Receiver Overflow errors to be Non-Fatal errors

Status: For the steppings affected, see the Summary Table of Changes.

BY11. Thermal Trip May be Asserted During Cold Reset

Problem: A false thermal-trip event can occur during cold reset.

Implication: Due to this erratum, the system may shutdown during cold reset.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY12. Writes to IA32_DEBUGCTL MSR May Fail when FREEZE_LBRS_ON_PMI is Set

Problem: When the FREEZE_LBRS_ON_PMI, IA32_DEBUGCTL MSR (1D9H) bit [11], is set, future writes to 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: For the steppings affected, see the Summary Table of Changes.

BY13. 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 0xFFFF_FFFF_FFFF_FFFF, both MSRs are supposed to synchronously reset to 0x0 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 rely on 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 wouldn't happen for over 10 years.

Workaround: None identified.

Status: For the steppings affected, see the Summary Table of Changes.

BY14. A Page Fault May Not be Generated When the PS bit is set to "1" in a PML4E or PDPTE

Problem: On processors supporting Intel® 64 architecture, the PS bit (Page Size, bit 7) is reserved in PML4Es and PDPTEs. If the translation of the linear address of a memory access encounters a PML4E or a PDPTE with PS set to 1, a page fault should occur. Due to this erratum, PS of such an entry is ignored and no page fault will occur due to its being set.

Implication: Software may not operate properly if it relies on the processor to deliver page faults when reserved bits are set in paging-structure entries.

Workaround: Software should not set bit 7 in any PML4E or PDPTE that has Present Bit (Bit 0) set to "1."

Status: For the steppings affected, see the Summary Table of Changes.

BY15. LBR/BTM/BTS Information Immediately After a Transition From Legacy/Compatibility Mode to 64-bit Mode May be Incorrect

Problem: If a transition from legacy/compatibility mode to 64-bit mode occurs and another branch event occurs before the first instruction executes (for example an external interrupt or trap) then any FROM address recorded by LBR (Last Branch Record), BTM (Branch Trace Message) or BTS (Branch Trace Store) on that second event may incorrectly report the upper 32 bits as zero.

Implication: Due to this erratum, bits 63:32 of the 'FROM' value for LBR/BTM/BTS may be improperly zeroed after a transition to 64 bite mode when the RIP (Instruction Pointer Register) is greater than 4 Gigabyte.

Workaround: None identified. This erratum may be detected by a 'FROM' address having its upper 32-bits zero but its lower 32-bits matching the previous 'TO' address recorded.

Status: For the steppings affected, see the Summary Table of Changes.

BY16. VID Information in IA32_PERF_STS MSR Bits [7:0] May be Incorrect

Problem: IA32_PERF_STS MSR (198H) bits [7:0] are supposed to indicate the VID (Voltage ID) after an Enhanced Intel SpeedStep® Technology transition. Due to this erratum, one core in a dual core CPU may report incorrect VID values in certain corner cases.

Implication: IA32_PERF_STS MSR bits [7:0] may contain incorrect VID values after certain Enhanced Intel SpeedStep® Technology transitions.

Workaround: None identified.

Status: For the steppings affected, see the Summary Table of Changes.

BY17. 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: For the steppings affected, see the Summary Table of Changes.

BY18. A VM Exit Occurring in IA-32e Mode May Not Produce a 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: For the steppings affected, see the Summary Table of Changes.

BY19. C6 Auto-Demotion Does Not Occur as Expected

Problem: C6 auto-demotion is enabled by setting PMG_CST_CONFIG_CONTROL MSR (E2H) bit 25 to 1. When enabled, C6 C-state is demoted to a lower C-state if average C6 residency time is low. Due to this erratum auto-demotion will not occur even when this bit is set and the average C6 residency time is low.

Implication: Due to this erratum the C6 transition may occur more frequently than desired.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY20. APIC Timer Can Expire Earlier Than Expected

Problem: Under certain circumstances, the APIC Timer may expire early on the first expiration after the CCR (Current Count Register) of the Local APIC is programmed. If the CCR in the APIC is written when BUS clock frequency is twice as fast as the nominal BUS clock, the first countdown of the APIC Timer may expire ***n*** nominal BUS clocks early, where ***n*** can be [1, 2, 4, 8, 16, 32, 64, 128] as determined by the DCR (Divide Configuration Register) in the Local APIC. In the worst case, for a nominal BUS frequency of 100 MHz and a DCR of 128 the APIC Timer can expire 1280 ns early.

Implication: When the CCR is written while BUS clock frequency is twice as fast as the nominal BUS clock the APIC timer may expire early on the first expiration.

Workaround: None identified.

Status: For the steppings affected, see the Summary Table of Changes.

BY21. Core C-state Residency MSR Values Are Incorrect

Problem: RDMSR of the core C-state residency MSRs will return a value of zero regardless of the actual residency time. This affects the following MSRs:

- C2_RESIDENCY_TIMER (3F8H)
- C4_RESIDENCY_TIMER (3F9H)
- C6_RESIDENCY_TIMER (3FAH)

Implication: Software cannot determine core C-state residencies by reading these MSRs.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY22. Local APIC Timer Expiration May Not be Honored if Software Requests Different Deep C-states For Each Logical Processor

Problem: If one logical processor requests the C6 state and the other logical processor requests the C4 state in close temporal proximity, the local APIC timer expiration event of the logical processor going to C6 may be dropped.

Implication: The platform will wake based only on the local APIC timer of the logical processor going to C4, or can hang if the timer is disabled.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY23. Task-switching IRET May Not Guarantee That All Memory Writes Have Become Globally Visible

Problem: Serializing instructions ensure that all preceding memory writes have become globally visible before any subsequent load or store instruction. An IRET is a serializing instruction. Due to this erratum, an IRET instruction that performs a task-switch does not guarantee serializing behavior.

Implication: Memory writes may not be observed in the correct order with respect to the task-switching IRET instruction.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY24. An Indirect Jump Instruction May Execute to An Incorrect Target

Problem: An indirect jump instruction may execute incorrectly to the target from a previous instance of the jump instead of to the correct target. This may occur only when the indirect jump is part of a short loop.

Implication: Due to this erratum, an indirect jump may cause unpredictable system behavior.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY25. A VM Exit as a Result of a Task-Switching IRET May Not Properly Save the State of NMI Blocking

Problem: A VM exit due to a fault or an EPT violation caused by an IRET instruction should set bit 12 (NMI unblocking due to IRET) of the VM-exit interruption-information field or the exit qualification respectively. This is required if IRET began execution when either (1) the “NMI exiting” VM-execution control was 0 and blocking by NMI was in effect; or (2) the “virtual NMIs” VM-execution control was 1 and virtual-NMI blocking was in effect. Due to this erratum, such VM exits may clear this bit if the IRET was causing a task switch.

Implication: If a VMM resumes a guest at the IRET instruction after handling such a VM exit, an NMI or a VM exit due to the 1-setting of the “NMI-window exiting” VM-execution control may be delivered prematurely.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY26. 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: For the steppings affected, see the Summary Table of Changes.

BY27. Software Requested or Thermal Based Clock Modulation May Not Result in an Accurate Stop-Clock Duty Cycle

Problem: Clock modulation can occur either by software request through the IA32_CLOCK_MODULATION MSR (19AH) or based on thermal conditions when the TM1 feature is enabled (bit 3) in IA32_MISC_ENABLE MSR(1A0H). Both engage internal stop-clock circuitry to impose a requested clocking duty cycle. This requested duty cycle ranges from 12.5% to 87.5%. Due to this erratum, the actual duty cycle may deviate from the requested duty cycle.

Implication: Software which is sensitive to the accuracy of the requested clock modulation duty cycle may not operate properly.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY28. Programming MSR_PKG_CST_CONFIG_CONTROL Package C-state Limit Field to 0x3 May Result in a System Hang

Problem: When software requests an MWAIT with a target state of 2 or greater and the MSR_PKG_CST_CONFIG_CONTROL MSR (0E2H) Package C-state Limit (bits 2:0) is programmed with a value of 0x3 it is possible for the system to hang.

Implication: Due to this erratum, an MWAIT instruction could result in a system hang.

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

Status: For the steppings affected, see the Summary Table of Changes.

BY29. CS Limit Violations May Not be Detected After VM Entry

Problem: The processor may fail to detect a CS limit violation on fetching the first instruction after VM entry if the first byte of that instruction is outside the CS limit but the last byte of the instruction is inside the limit.

Implication: The processor may erroneously execute an instruction that should have caused a general protection exception.

Workaround: When a VMM emulates a branch instruction, it should inject a general protection exception if the instruction's target EIP is beyond the CS limit.

Status: For the steppings affected, see the Summary Table of Changes.

BY30. Fast-String Operations Are Not Enabled by Default

Problem: IA32_MISC_ENABLE MSR (1A0H) Fast Strings Enable bit 0 should be set to 1 after reset. Due to this erratum, it has a value of 0 after reset.

Implication: Code capable of utilizing fast-string operations will not be optimized until fast strings are enabled.

Workaround: Software can enable the fast-strings feature by setting bit 0 of IA32_MISC_ENABLE MSR.

Status: For the steppings affected, see the Summary Table of Changes.

BY31. PCIe* Root Ports May Incorrectly Indicate CRS Software Visibility Support

Problem: The PCIe Root Port ROOTCAP.CRSSV (Bus 0; Device 1-4; Function 0; offset 5EH; bit 0) field indicates that the root port is capable of returning CRS (Configuration Request Retry Status) completion status to software. Due to this erratum, the default value of this bit is set to 1, incorrectly indicating that CRS is supported.

Implication: Due to this erratum, software that expects CRS completion status may not function as expected.

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

Status: For the steppings affected, see the Summary Table of Changes.

§

Specification Changes

There are no specification changes at this time.

§

Specification Clarifications

There are no specification clarifications at this time.

§

Documentation Changes

There are no documentation changes in this specification update revision.

§