Intel® Xeon® E7-8800/4800 v4 Processor Product Family

Specification Update

May 2023

Order Number: 334163-022US

Intel technologies features and benefits depend on system configuration and may require enabled hardware, software, or service activation. Learn more at intel.com, or from the OEM or retailer.

No computer system can be absolutely secure. Intel does not assume any liability for lost or stolen data or systems or any damages resulting from such losses.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

Intel® Turbo Boost Technology requires a PC with a processor with Intel® Turbo Boost Technology capability. Intel® Turbo Boost Technology performance varies depending on hardware, software and overall system configuration. Check with your PC manufacturer on whether your system delivers Intel® Turbo Boost Technology. For more information, see http://www.intel.com/technology/turboboost.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting www.intel.com/design/literature.htm.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

Contents

Revision History

Document Number	Revision	Description	Date
334165	022US	- Added erratum BDEX11	May 2023
334165	021	- Added erratum BDX103	February 2021
334165	020	- Added erratum BDEX10	January 2021
334165	019	- Added erratum BDX102	October 2020
334165	018	- Added erratum BDX101	July 2020
334165	017	- Added erratum BDX100	November 2019
334165	016	- Added errata BDX98 - BDX99	August 2019
334165	015	- Added erratum BDX97	July 2019
334165	014	- Updated BDX76	April 2019
334165	013	- Added errata BDX93 - BDX96	June 2018
334165	012	- Updated erratum BDF88	April 2018
334165	011	- Added erratum BDEX9	January 2018
334165	010	- Added errata BDX91- BDX92	September 2017
334165	009	- Added erratum BDX90 - Added Specification Change SCh1	April 2017
334165	008	- Added erratum BDX89	March 2017
334165	007	- Updated erratum BDX41, BDX68	January 2017
334165	006	- Added errata BDX87 - BDX88 - Added erratum BDEX8	December 2016
334165	005	- Added errata BDX84 - BDX86 - Updated errata names BDF -> BDX - Updated errata names EX -> BDEX	November 2016
334165	004	- Removed BDF69 due to inapplicability - Removed BDF70, duplicate of BDF72 - Added errata BDF75 - BDF83, EX6 - EX7	October 2016
334165	003	- Added errata BDF72 - BDF74	September 2016
334165	002	- Added BDF70 & BDF71	August 2016
334165	001	- Initial release	July 2016

Preface

This document is an update to the specifications contained in the Related Documents table below. This document is a compilation of device and documentation sighting, specification clarifications and changes. It is intended for hardware system manufacturers and software developers of applications, operating systems, or tools.

Information types defined in are consolidated into this document and are no longer published in other documents.

This document may also contain information that was not previously published.

Related Documents

Document Title	Document Number/ Location
Intel® 64 and IA-32 Architectures Software Developer's Manual Combined Volumes: 1, 2A, 2B, 2C, 2D, 3A, 3B, 3C, 3D, and 4 • **Volume 1:** Describes the architecture and programming environment of processors supporting IA-32 and Intel® 64 architectures. • **Volume 2:** Includes the full instruction set reference, A-Z. Describes the format of the instruction and provides reference pages for instructions. • **Volume 3:** Includes the full system programming guide, parts 1, 2, 3, and 4. Describes the operating-system support environment of Intel® 64 and IA-32 architectures, including: memory management, protection, task management, interrupt and exception handling, multi-processor support, thermal and power management features, debugging, performance monitoring, system management mode, Virtual Machine Extensions (VMX) instructions, Intel® Virtualization Technology (Intel® VT), and Intel® Software Guard Extensions (Intel® SGX). • **Volume 4:** Describes the model-specific registers of processors supporting IA-32 and Intel® 64 architectures.	https://software.intel.com/content/www/us/en/develop/download/intel-64-and-ia-32-architectures-sdm-combined-volumes-1-2a-2b-2c-2d-3a-3b-3c-3d-and-4.html?wapkw=Intel%C2%AE%2064%20and%20IA-32%20Architecture%20Software%20Developer%E2%80%99s%20Manual

Nomenclature

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

QDF Number is a four digit code used to distinguish between engineering samples. These samples are used for qualification and early design validation. The functionality of these parts can range from mechanical only to fully functional. This document has a processor identification information table that lists these QDF numbers and the corresponding product details.

Known Sample Issues are known issues with samples that are root caused and dispositioned to design defects or errors. A known sample issue may cause the behavior of the Intel® Xeon® E7-8800/4800 v4 Processor Product Family samples to deviate from published specifications.

Hardware and software designed to be used with any given stepping must assume that all known sample issues documented for that stepping are present in all devices.

Sightings are design defects or errors. These may cause the Intel® Xeon® E7-8800/4800 v4 Processor Product Family behavior to deviate from published specifications. Hardware and software designed to be used with any given stepping must assume that all sightings documented for that stepping are present on all devices.

Closed Non-Si Sightings are closed sightings that are resolved to be not related to the Intel® Xeon® E7-8800/4800 v4 Processor Product Family. These include issues that may be third-party issues, test configuration issues, documentation issues, board issues, and so forth.

The sighting/known issue numbers contained in this document apply to Intel's internal issue tracking system and have no bearing on the overall number of issues with this project. All issues identified in this report only apply to engineering sample silicon steppings. Intel intends to fix these issues before releasing the production silicon unless otherwise indicated. Therefore, these sightings/known issues may not have any impact on the production steppings of the components.

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

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 any new release of the specification.

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

Note: Specification changes, specification clarifications and documentation changes are removed from the sightings report and/or specification update when the appropriate changes are made to the appropriate product specification or user documentation.

§

Identification Information

Component Identification via Programming Interface

The Intel® Xeon® Processor E7-8800/4800 v4 Product Family stepping can be identified by the following register contents:

Reserved	Extended Family[1]	Extended Model[2]	Reserved	Processor Type[3]	Family Code[4]	Model Number[5]	Stepping ID[6]
31:28	27:20	19:16	15:14	13:12	11:8	7:4	3:0
	000000000	b100		000	b110	b1111	Varies with Stepping

Notes:

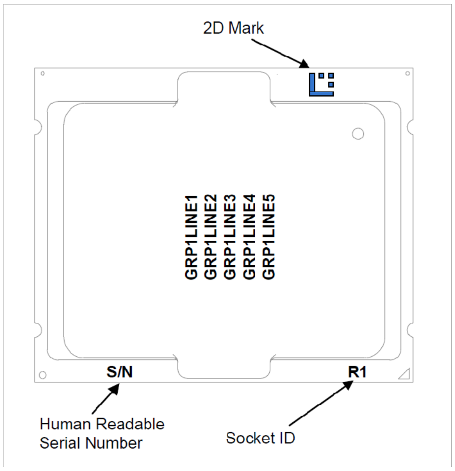
1. The Extended Family, Bits [27:20] are used in conjunction with the Family Code, specified in Bits [11:8], to indicate whether the processor belongs to the Intel386™, Intel486™, Pentium®, Pentium® 4, or Intel® Core™ processor family.
2. The Extended Model, Bits [19:16] in conjunction with the Model Number, specified in Bits [7:4], are used to identify the model of the processor within the processor's family.
3. The Family Code corresponds to Bits [11:8] of the Extended Data Register (EDX) register after RESET, Bits [11:8] of the Extended Accumulator Register (EAX) register after the CPU Identification (CPUID) instruction is executed with a 1 in the EAX register, and the generation field of the Device ID register accessible through Boundary Scan.
4. The Model Number corresponds to Bits [7:4] of the EDX register after RESET, Bits [7:4] of the EAX register after the CPUID instruction is executed with a 1 in the EAX register, and the model field of the Device ID register accessible through Boundary Scan.
5. The Stepping ID in Bits [3:0] indicates the revision number of that model. See Appendix 1 for the processor stepping ID number in the CPUID information.

When EAX is initialized to a value of "1", the CPUID instruction returns the Extended Family, Extended Model, Processor Type, Family Code, Model Number and Stepping ID value in the EAX register. Notice that the EDX processor signature value after reset is equivalent to the processor signature output value in the EAX register.

The Cache and Transaction Lookaside Buffers (TLB) descriptor parameters are provided in the EAX, the Extended Base Register (EBX), the Extended Count Register (ECX), and the EDX registers after the CPUID instruction is executed with a 2 in the EAX register.

Component Marking Information

Figure 1. Intel® Xeon® E7-8800/4800 v4 Processor Product Family Top-side Markings (Example)



Legend:	Mark Text (Production Mark):
GRP1LINE1:	i{M}{C}YY
GRP1LINE2:	SUB-BRAND PROC#
GRP1LINE3:	SSPEC SPEED
GRP1LINE4:	XXXXX
GRP1LINE5:	{FPO} {e4}

For the Intel® Xeon® E7-8800/4800 v4 Processor Product Family SKUs see https://ark.intel.com/content/www/us/en/ark/products/series/93797/intel-xeon-processor-e7-v4-family.html.

§

Summary Tables of Changes

The following tables indicate the errata, specification changes, specification clarifications, or documentation changes, which apply to the Product Name product. Intel may fix some of the errata in a future stepping of the component, and account for the other outstanding issues through documentation or specification changes as noted. These tables use the following notations:

Codes Used in Summary Tables

Stepping

X: Errata exist in the stepping indicated. 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.

Page

(Page): Page location of item in this document.

Status

Doc: Document change or update 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

▎ Change bar to left of table row indicates this erratum is either new or modified from the previous version of the document.

Table 1. Intel® Xeon® Processor E5-2600 and E7-8800/4800 Shared Integrated Core/ Uncore Errata (Sheet 1 of 4)

Number	Steppings	Status	ERRATA
	B0		
BDX1	X	No Fix	Enabling ISOCH Mode May Cause The System to Hang
BDX2	X	No Fix	PCI BARs in the Home Agent Will Return Non-Zero Values During Enumeration
BDX3	X	No Fix	PCIe* Header of a Malformed TLP Is Logged Incorrectly
BDX4	X	No Fix	A Malformed TLP May Block ECRC Error Logging
BDX5	X	No Fix	The System May Hang During an Intel® QuickPath Interconnect (Intel® QPI) Slow to Fast Mode Transition
BDX6	X	No Fix	Unexpected Performance Loss When Turbo Disabled
BDX7	X	No Fix	Exiting From Package C3 or Package C6 With DDR4-2133 May Lead to Unpredictable System Behavior
BDX8	X	No Fix	The System May Shut Down Unexpectedly During a Warm Reset
BDX9	X	No Fix	CAT May Not Behave as Expected
BDX10	X	No Fix	LBR, BTS, BTM May Report a Wrong Address When an Exception/Interrupt Occurs in 64-bit Mode
BDX11	X	No Fix	EFLAGS Discrepancy on Page Faults and on EPT-Induced VM Exits after a Translation Change
BDX12	X	No Fix	MCi_Status Overflow Bit May Be Incorrectly Set on a Single Instance of a DTLB Error
BDX13	X	No Fix	LER MSRs May Be Unreliable
BDX14	X	No Fix	MONITOR or CLFLUSH on the Local XAPIC's Address Space Results in Hang
BDX15	X	No Fix	#GP on Segment Selector Descriptor that Straddles Canonical Boundary May Not Provide Correct Exception Error Code
BDX16	X	No Fix	FREEZE_WHILE_SMM Does Not Prevent Event From Pending PEBS During SMM
BDX17	X	No Fix	APIC Error “Received Illegal Vector” May Be Lost
BDX18	X	No Fix	Performance Monitor Precise Instruction Retired Event May Present Wrong Indications
BDX19	X	No Fix	CR0.CD Is Ignored in VMX Operation
BDX20	X	No Fix	Instruction Fetch May Cause Machine Check if Page Size and Memory Type Was Changed Without Invalidation
BDX21	X	No Fix	Execution of VAESIMC or VAESKEYGENASSIST With An Illegal Value for VEX.vvvv May Produce a #NM Exception
BDX22	X	No Fix	Interrupt From Local APIC Timer May Not Be Detectable While Being Delivered
BDX23	X	No Fix	Pending x87 FPU Exceptions (#MF) May Be Signaled Earlier Than Expected
BDX24	X	No Fix	DR6.B0-B3 May Not Report All Breakpoints Matched When a MOV/POP SS Is Followed by a Store or an MMX Instruction
BDX25	X	No Fix	VEX.L Is Not Ignored with VCVT*2SI Instructions
BDX26	X	No Fix	Processor May Livelock During On Demand Clock Modulation
BDX27	X	No Fix	Performance Monitor Events OTHER_ASSISTS.AVX_TO_SSE And OTHER_ASSISTS.SSE_TO_AVX May Over Count
BDX28	X	No Fix	Performance Monitor Event DSB2MITE_SWITCHES.COUNT May Over Count
BDX29	X	No Fix	Timed MWAIT May Use Deadline of a Previous Execution
BDX30	X	No Fix	IA32_VMX_VMCS_ENUM MSR (48AH) Does Not Properly Report The Highest Index Value Used For VMCS Encoding
BDX31	X	No Fix	Incorrect FROM_IP Value For an RTM Abort in BTM or BTS May Be Observed

Table 1. Intel® Xeon® Processor E5-2600 and E7-8800/4800 Shared Integrated Core/Uncore Errata (Sheet 2 of 4)

Number	Steppings	Status	ERRATA
	B0		
BDX32	X	No Fix	Locked Load Performance Monitoring Events May Under Count
BDX33	X	No Fix	Transactional Abort May Cause an Incorrect Branch Record
BDX34	X	No Fix	PMI May be Signaled More Than Once For Performance Monitor Counter Overflow
BDX35	X	No Fix	Execution of FXSAVE or FXRSTOR With the VEX Prefix May Produce a #NM Exception
BDX36	X	No Fix	VM Exit May Set IA32_EFER.NXE When IA32_MISC_ENABLE Bit 34 Is Set to 1
BDX37	X	No Fix	A MOV to CR3 When EPT Is Enabled May Lead to an Unexpected Page Fault or an Incorrect Page Translation
BDX38	X	No Fix	Intel® Processor Trace (Intel® PT) Packet Generation May Stop Sooner Than Expected
BDX39	X	No Fix	PEBS Eventing IP Field May Be Incorrect After Not-Taken Branch
BDX40	X	No Fix	Reading The Memory Destination of an Instruction That Begins an HLE Transaction May Return the Original Value
BDX41	X	No Fix	Removed Due to Inapplicability
BDX42	X	No Fix	Performance Monitoring Event INSTR_RETIRED.ALL May Generate Redundant PEBS Records For an Overflow
BDX43	X	No Fix	Reset During PECI Transaction May Cause a Machine Check Exception
BDX44	X	No Fix	Intel® PT MODE.Exec, PIP, and CBR Packets Are Not Generated as Expected
BDX45	X	No Fix	Performance Monitor Instructions Retired Event May Not Count Consistently
BDX46	X	No Fix	General-Purpose Performance Counters May be Inaccurate with Any Thread
BDX47	X	No Fix	An Invalid LBR May Be Recorded Following a Transactional Abort
BDX48	X	No Fix	Executing an RSM Instruction With Intel® PT Enabled Will Signal a #GP
BDX49	X	No Fix	Intel PT PIP May be Unexpectedly Generated
BDX50	X	No Fix	Processor Core Ratio Changes While in Probe Mode May Result in a Hang
BDX51	X	No Fix	Processor Does Not Check IRTE Reserved Bits
BDX52	X	No Fix	PCIe* TPH Request Capability Structure Incorrectly Advertises Device Specific Mode as Supported
BDX53	X	No Fix	Package C3 State or Deeper May Lead to a Reset
BDX54	X	No Fix	VMX-Preemption Timer May Stop Operating When ACC Is Enabled
BDX55	X	No Fix	Intel® Advanced Vector Extensions (Intel® AVX) Workloads May Exceed ICCMAX Limits
BDX56	X	No Fix	Writing MSR_ERROR_CONTROL May Cause a #GP
BDX57	X	No Fix	Enabling ACC in VMX Non-Root Operation May Cause System Instability
BDX58	X	No Fix	A Spurious Patrol Scrub Error May Be Logged
BDX59	X	No Fix	Performance Monitoring Counters May Produce Incorrect Results for BR_INST_RETIRED Event on Logical Processor
BDX60	X	No Fix	An APIC Timer Interrupt During Core C6 Entry May Be Lost
BDX61	X	No Fix	Processor Instability May Occur When Using The PECI RdIAMSR Command
BDX62	X	No Fix	A #VE May Not Invalidate Cached Translation Information
BDX63	X	No Fix	Package C-state Transitions While Inband PECI Accesses Are in Progress May Cause Performance Degradation
BDX64	X	No Fix	Attempting Concurrent Enabling of Intel® PT With LBR, BTS, or BTM Results in a #GP

Table 1. Intel® Xeon® Processor E5-2600 and E7-8800/4800 Shared Integrated Core/ Uncore Errata (Sheet 3 of 4)

Number	Steppings	Status	ERRATA
	B0		
BDX65	X	No Fix	A DDR4 C/A Parity Error in Lockstep Mode May Result in a Spurious Uncorrectable Error
BDX66	X	No Fix	Cores May be Unable to Reach Maximum Turbo Frequency
BDX67	X	No Fix	PEBS Record May Be Generated After Being Disabled
BDX68	X	No Fix	Removed Due to Duplication of BDX76
BDX69	X	No Fix	Removed Due to Inapplicability
BDX70	X	No Fix	Removed Due to Duplication of BDX72
BDX71	X	No Fix	PEBS EventingIP Field May Be Incorrect Under Certain Conditions
BDX72	X	No Fix	Turbo May Be Delayed After Exiting C6 When Using HWP
BDX73	X	No Fix	Writing The IIO_LLC_WAYS MSR Results in an Incorrect Value
BDX74	X	No Fix	RF May Be Incorrectly Set in the EFLAGS That Is Saved on a Fault in PEBS or BTS
BDX75	X	No Fix	The System May Hang When Executing a Complex Sequence of Locked Instructions
BDX76	X	No Fix	Using Intel® TSX Instructions May Lead to Unpredictable System Behavior
BDX77	X	No Fix	Data Breakpoint Coincident With a Machine Check Exception May be Lost
BDX78	X	No Fix	Internal Parity Errors May Incorrectly Report Overflow in the IA32_MC0_STATUS MSR
BDX79	X	No Fix	Incorrect VMCS Used for PML-Index field on VMX Transitions Into and Out of SMM
BDX80	X	No Fix	An APIC Timer Interrupt During Core C6 Entry May Be Lost
BDX81	X	No Fix	Inband PECI Concurrent With OS Patch Load May Result in Incorrect Throttling Causing Reduced System Performance
BDX82	X	No Fix	An Intel® Hyper-Threading Technology (Intel® HT Technology) Enabled Processor May Exhibit Internal Parity Errors or Unpredictable System Behavior
BDX83	X	No Fix	IA32_MC4_STATUS.VAL May be Incorrectly Cleared by Warm Reset
BDX84	X	No Fix	Some DRAM And L3 Cache Performance Monitoring Events May Undercount
BDX85	X	No Fix	An x87 Store Instruction Which Pends #PE While EPT Is Enabled May Lead to an Unexpected Machine Check and/or Incorrect x87 State Information
BDX86	X	No Fix	Load Latency Performance Monitoring Facility May Stop Counting
BDX87	X	No Fix	General-Purpose Performance Monitoring Counters 4-7 Will Not Increment Do Not Count With USR Mode Only Filtering
BDX88	X	No Fix	Writing MSR_LASTBRANCH_x_FROM_IP and MSR_LER_FROM_LIP May #GP When Intel® TSX Is Not Supported
BDX89	X	No Fix	APIC Timer Interrupt May Not be Generated at the Correct Time In TSC-Deadline Mode
BDX90	X	No Fix	Loading Microcode Updates or Executing an Authenticated Code Module May Result in a System Hang
BDX91	X	No Fix	NVDIMM Data May Not be Preserved Correctly on Power Loss or ADR Activation
BDX92	X	No Fix	Link Down Events Behind PCIe* Device Connected to CPU Root Ports Can Cause CTO > 50ms on Other Root Ports
BDX93	X	No Fix	Reads From MSR_LER_TO_LIP May Not Return a Canonical Address
BDX94	X	No Fix	Processor May Hang After Multiple Microcode Updates Loaded
BDX95	X	No Fix	In eMCA2 Mode, When the Retirement Watchdog Timeout Occurs CATERR# May be Asserted
BDX96	X	No Fix	Systems That Enable Both OSB and IODC May Exhibit Unexpected System Behavior
BDX97	X	No Fix	MBM Counters May Report System Memory Bandwidth Incorrectly

Table 1. Intel® Xeon® Processor E5-2600 and E7-8800/4800 Shared Integrated Core/ Uncore Errata (Sheet 4 of 4)

Number	Steppings	Status	ERRATA
	B0		
BDX98	X	No Fix	When Operating at Maximum Turbo Frequencies, The Processor May Hang
BDX99	X	No Fix	A Pending Fixed Interrupt May Be Dispatched Before an Interrupt of The Same Priority Completes
BDX100	X	No Fix	Instruction Fetch May Cause Machine Check if Page Size Was Changed Without Invalidation
BDX101	X	No Fix	Overflow Flag in IA32_MC0_STATUS MSR May Be Incorrectly Set
BDX102	X	No Fix	PECI RdIAMSR() to Non-existent MSR Might Result in IERR
BDX103	X	No Fix	Retried PECI PCIConfigLocal Register Accesses May Not Operate Correctly

Specification Changes

Number	SPECIFICATION CHANGES
1	None for this revision of this specification update.

Specification Clarifications

No.	SPECIFICATION CLARIFICATIONS
1	None for this revision of this specification update.

Documentation Changes

No.	DOCUMENTATION CHANGES
1	None for this revision of this specification update.

Intel® Xeon® E5-2600 and E7-8800/4800 Processors Shared Integrated Core/Uncore Errata

BDX1 Enabling ISOCH Mode May Cause The System to Hang

Problem: When Isochronous (ISOCH) operation is enabled within BIOS, the system may hang and fail to boot.

Implication: Due to this erratum, the system may hang and fail to boot.

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

Status: For the steppings affected, see the Table 1.

BDX2 PCI BARs in the Home Agent Will Return Non-Zero Values During Enumeration

Problem: During system initialization the Operating System may access the standard Peripheral Component Interconnect (PCI) Base Address Registers (BARs). Due to this erratum, accesses to the Home Agent BAR registers (Bus 1; Device 18; Function 0,4; Offsets 0x14-0x24) will return non-zero values.

Implication: The operating system may issue a warning. Intel has not observed any functional failures due to this erratum.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX3 PCIe* Header of a Malformed TLP Is Logged Incorrectly

Problem: If a PCIe* port receives a malformed Transaction Layer Packet (TLP), an error is logged in the UNCERRSTS register (Device 0; Function 0; Offset 14CH and Device 2-3; Function 0-3; Offset 14CH). Due to this erratum, the header of the malformed TLP is logged incorrectly in the HDRLOG register (Device 0; Function 0; Offset 164H and Device 2-3; Function 0-3; Offset 164H).

Implication: The PCIe* header of a malformed TLP is not logged correctly.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX4 A Malformed TLP May Block ECRC Error Logging

Problem: If a PCIe* port receives a Malformed TLP that also would generate an End-to-End CRC (ECRC) Check Failed error, it should report a Malformed TLP error. When Malformed TLP errors are masked, the processor should report the lower-precedence ECRC Check Failed error but, due to this erratum, it does not.

Implication: Software that relies upon ECRC Check Failed error indication may not behave as expected.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX5 The System May Hang During an Intel® QuickPath Interconnect (Intel® QPI) Slow to Fast Mode Transition

Problem: During an Intel® QPI slow mode to fast mode transition, the LL_STATUS field of the QPIPCSTS register (Bus 0; Device 8,9,10; Function 0; Offset 0xc0) may not be correctly updated to reflect link readiness.

Implication: The system may hang waiting for the QPIPCSTS.LL_STATUS to update.

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

Status: For the steppings affected, see the Table 1.

BDX6 Unexpected Performance Loss When Turbo Disabled

Problem: When Intel® Turbo Boost Technology is disabled by IA32_MISC_ENABLES MSR (416H) TURBO_MODE_DISABLE bit 38, the Ring operating frequency may be below P1 operating frequency.

Implication: Processor performance may be below expectations for P1 operating frequency.

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

Status: For the steppings affected, see the Table 1.

BDX7 Exiting From Package C3 or Package C6 With DDR4-2133 May Lead to Unpredictable System Behavior

Problem: Due to this erratum, with DDR4-2133 memory, exiting from Package C3 (PC3) or Package C6 (PC6) state may lead to unpredictable system behavior.

Implication: This erratum may lead to unpredictable system behavior.

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

Status: For the steppings affected, see the Table 1.

BDX8 The System May Shut Down Unexpectedly During a Warm Reset

Problem: Certain complex internal timing conditions present when a warm reset is requested can prevent the orderly completion of in-flight transactions. It is possible under these conditions that the warm reset will fail and trigger a full system shutdown.

Implication: When this erratum occurs, the system will shut down and all machine check error logs will be lost.

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

Status: For the steppings affected, see the Table 1.

BDX9 CAT May Not Behave as Expected

Problem: Due to this erratum, Cache Allocation Technology (CAT) way enforcement may not behave as configured.

Implication: When this erratum occurs, cache quality of service guarantees may not be met.

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

Status: For the steppings affected, see the Table 1.

BDX10 LBR, BTS, BTM May Report a Wrong Address When an Exception/ Interrupt Occurs in 64-bit Mode

Problem: An exception/interrupt event should be transparent to the Last Branch Record (LBR), Branch Trace Store (BTS) and Branch Trace Message (BTM) mechanisms. However, during a specific boundary condition where the exception/interrupt occurs right after the execution of an instruction at the lower canonical boundary (0x00007FFFFFFFFFFF) in 64-bit mode, the LBR return registers will save a wrong return address with bits 63 to 48 incorrectly sign extended to all 1's. Subsequent BTS and BTM operations which report the LBR will also be incorrect.

Implication: LBR, BTS and BTM may report incorrect information in the event of an exception/interrupt.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX11 EFLAGS Discrepancy on Page Faults and on EPT-Induced VM Exits after a Translation Change

Problem: EFLAGS Discrepancy on Page Faults and on EPT-Induced VM Exits after a Translation Change.

This erratum is regarding the case where paging structures are modified to change a linear address from writable to non-writable without software performing an appropriate Translation Lookaside Buffer (TLB) invalidation. When a subsequent access to that address by a specific instruction (ADD, AND, BTC, BTR, BTS, CMPXCHG, DEC, INC, NEG, NOT, OR, ROL/ROR, SAL/SAR/SHL/SHR, SHLD, SHRD, SUB, XOR, and XADD) causes a page fault or an EPT-induced VM exit, the value saved for EFLAGS may incorrectly contain the arithmetic flag values that the EFLAGS register would have held had the instruction completed without fault or VM exit. For page faults, this can occur even if the fault causes a VM exit or if its delivery causes a nested fault.

Implication: None identified. Although the EFLAGS value saved by an affected event (a page fault or an EPT-induced VM exit) may contain incorrect arithmetic flag values, Intel has not identified software that is affected by this erratum. This erratum will have no further effects once the original instruction is restarted because the instruction will produce the same results as if it had initially completed without fault or VM exit.

Workaround: If the handler of the affected events inspects the arithmetic portion of the saved EFLAGS value, then system software should perform a synchronized paging structure modification and TLB invalidation.

Status: For the steppings affected, see the Table 1.

BDX12 MCi_Status Overflow Bit May Be Incorrectly Set on a Single Instance of a DTLB Error

Problem: A single Data Translation Look Aside Buffer (DTLB) error can incorrectly set the Overflow (bit [62]) in the `MCi_Status` register. A DTLB error is indicated by Media Access Control (MCA) error code (bits [15:0]) appearing as binary value, 000x 0000 0001 0100, in the `MCi_Status` register.

Implication: Due to this erratum, the Overflow bit in the `MCi_Status` register may not be an accurate indication of multiple occurrences of DTLB errors. There is no other impact to normal processor functionality.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX13 LER MSRs May Be Unreliable

Problem: Due to certain internal processor events, updates to the Last Exception Record (LER) MSRs, `MSR_LER_FROM_LIP` (1DDH) and `MSR_LER_TO_LIP` (1DEH), may happen when no update was expected.

Implication: The values of the LER MSRs may be unreliable.

Workaround: None Identified.

Status: For the steppings affected, see the Table 1.

BDX14 MONITOR or CLFLUSH on the Local XAPIC's Address Space Results in Hang

Problem: If the target linear address range for a MONITOR or CLFLUSH is mapped to the local xAPIC's address space, the processor will hang.

Implication: When this erratum occurs, the processor will hang. The local xAPIC's address space must be uncached. The MONITOR instruction only functions correctly if the specified linear address range is of the type write-back. CLFLUSH flushes data from the cache. Intel has not observed this erratum with any commercially available software.

Workaround: Do not execute MONITOR or CLFLUSH instructions on the local xAPIC address space.

Status: For the steppings affected, see the Table 1.

BDX15 #GP on Segment Selector Descriptor that Straddles Canonical Boundary May Not Provide Correct Exception Error Code

Problem: During a General Protection (#GP) Exception, the processor pushes an error code on to the exception handler’s stack. If the segment selector descriptor straddles the canonical boundary, the error code pushed onto the stack may be incorrect.

Implication: An incorrect error code may be pushed onto the stack. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX16 FREEZE_WHILE_SMM Does Not Prevent Event From Pending PEBS During SMM

Problem: In general, a Precise-Event-Based-Sampling (PEBS) record should be generated on the first count of the event after the counter has overflowed. However, `IA32_DEBUGCTL_MSR.FREEZE_WHILE_SMM` (MSR 1D9H, bit [14]) prevents performance counters from counting during System Management Mode (SMM). Due to this erratum, if:

1. A performance counter overflowed before an System Management Interrupt (SMI).
2. A PEBS record has not yet been generated because another count of the event has not occurred.
3. The monitored event occurs during SMM then a PEBS record will be saved after the next Resume System Management Mode (RSM) instruction. When `FREEZE_WHILE_SMM` is set, a PEBS should not be generated until the event occurs outside of SMM.

Implication: A PEBS record may be saved after an RSM instruction due to the associated performance counter detecting the monitored event during SMM; even when `FREEZE_WHILE_SMM` is set.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX17 APIC Error “Received Illegal Vector” May Be Lost

Problem: Advanced Programmable Interrupt Controller (APIC) may not update the Error Status Register (ESR) flag Received Illegal Vector bit [6] properly when an illegal vector error is received on the same internal clock that the ESR is being written (as part of the write-read ESR access flow). The corresponding error interrupt will also not be generated for this case.

Implication: Due to this erratum, an incoming illegal vector error may not be logged into ESR properly and may not generate an error interrupt.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX18 Performance Monitor Precise Instruction Retired Event May Present Wrong Indications

Problem: When the Precise Distribution for Instructions Retired (PDIR) mechanism is activated (`INST_RETIRED.ALL` (event C0H, umask value 00H) on Counter 1 programmed in PEBS mode), the processor may return wrong PEBS/PMI interrupts and/or incorrect counter values if the counter is reset with a SAV below 100 (Sample-After-Value is the counter reset value software programs in `MSR IA32_PMC1[47:0]` in order to control interrupt frequency)**.**

Implication: Due to this erratum, when using low SAV values, the program may get incorrect PEBS or PMI interrupts and either or an invalid counter state.

Workaround: The sampling driver should avoid using SAV<100.

Status: For the steppings affected, see the Table 1.

BDX19 CR0.CD Is Ignored in VMX Operation

Problem: If CR0.CD=1, the MTRRs and PAT should be ignored and the UC memory type should be used for all memory accesses. Due to this erratum, a logical processor in Virtual Machine Extension (VMX) operation will operate as if CR0.CD=0 even if that bit is set to 1.

Implication: Algorithms that rely on cache disabling may not function properly in VMX operation.

Workaround: Algorithms that rely on cache disabling should not be executed in VMX root operation.

Status: For the steppings affected, see the Table 1.

BDX20 Instruction Fetch May Cause Machine Check if Page Size and Memory Type Was Changed Without Invalidation

Problem: This erratum may cause a machine-check error (`IA32_MCi_STATUS.MCACOD=0150H`) on the fetch of an instruction that crosses a 4-KByte address boundary. It applies only if (1) the 4-KByte linear region on which the instruction begins is originally translated using a 4-KByte page with the WB memory type; (2) the paging structures are later modified so that linear region is translated using a large page (2-MByte, 4-MByte, or 1-GByte) with the UC memory type; and (3) the instruction fetch occurs after the paging-structure modification but before software invalidates any TLB entries for the linear region.

Implication: Due to this erratum, an unexpected machine check with error code 0150H may occur, possibly resulting in a shutdown. Intel has not observed this erratum with any commercially available software.

Workaround: Software should not write to a paging-structure entry in a way that would change, for any linear address, both the page size and the memory type. It can instead use the following algorithm: first clear the P flag in the relevant paging-structure entry (for example, PDE); then invalidate any translations for the affected linear addresses; and then modify the relevant paging-structure entry to set the P flag and establish the new page size and memory type.

Status: For the steppings affected, see the Table 1.

BDX21 Execution of VAESIMC or VAESKEYGENASSIST With An Illegal Value for VEX.vvvv May Produce a #NM Exception

Problem: The VAESIMC and VAESKEYGENASSIST instructions should produce an Invalid-Opcode (#UD) exception if the value of the vvvv field in the VEX prefix is not 1111b. Due to this erratum, if CR0.TS is "1", the processor may instead produce a Device-Not-Available (#NM) exception.

Implication: Due to this erratum, some undefined instruction encodings may produce a #NM instead of a #UD exception.

Workaround: Software should always set the vvvv field of the VEX prefix to 1111b for instances of the VAESIMC and VAESKEYGENASSIST instructions.

Status: For the steppings affected, see the Table 1.

BDX22 Interrupt From Local APIC Timer May Not Be Detectable While Being Delivered

Problem: If the local-APIC timer's Current-Count Register (CCR) is 0, software should be able to determine whether a previously generated timer interrupt is being delivered by first reading the delivery-status bit in the Local Vector Table (LVT) timer register and then reading the bit in the Interrupt-Request Register (IRR) corresponding to the vector in the LVT timer register. If both values are read as 0, no timer interrupt should be in the process of being delivered. Due to this erratum, a timer interrupt may be delivered even if the CCR is 0 and the LVT and IRR bits are read as 0. This can occur only if the Divide Configuration Register (DCR) is greater than or equal to 4. The erratum does not occur if software writes zero to the Initial Count Register before reading the LVT and IRR bits.

Implication: Software that relies on reads of the LVT and IRR bits to determine whether a timer interrupt is being delivered may not operate properly.

Workaround: Software that uses the local-APIC timer must be prepared to handle the timer interrupts, even those that would not be expected based on reading CCR and the LVT and IRR bits; alternatively, software can avoid the problem by writing zero to the Initial Count Register before reading the LVT and IRR bits.

Status: For the steppings affected, see the Table 1.

BDX23 Pending x87 FPU Exceptions (#MF) May Be Signaled Earlier Than Expected

Problem: x87 instructions that trigger #MF normally service interrupts before the #MF. Due to this erratum, if an instruction that triggers #MF is executed while Enhanced Intel SpeedStep® Technology transitions, Intel® Turbo Boost Technology transitions, or Thermal Monitor events occur, the pending #MF may be signaled before pending interrupts are serviced.

Implication: Software may observe #MF being-signaled before pending interrupts are serviced.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX24 DR6.B0-B3 May Not Report All Breakpoints Matched When a MOV/POP SS Is Followed by a Store or an MMX Instruction

Problem: Normally, data breakpoints matches that occur on a MOV SS, r/m or POP SS will not cause a debug exception immediately after MOV/POP SS but will be delayed until the instruction boundary following the next instruction is reached. After the debug exception occurs, DR6.B0-B3 bits will contain information about data breakpoints matched during the MOV/POP SS as well as breakpoints detected by the following instruction. Due to this erratum, DR6.B0-B3 bits may not contain information about data breakpoints matched during the MOV/POP SS when the following instruction is either an MMX instruction that uses a memory addressing mode with an index or a store instruction.

Implication: When this erratum occurs, DR6 may not contain information about all breakpoints matched. This erratum will not be observed under the recommended usage of the MOV SS,r/m or POP SS instructions (that is, following them only with an instruction that writes (E/R)SP).

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX25 VEX.L Is Not Ignored with VCVT*2SI Instructions

Problem: The VEX.L bit should be ignored for the VCVT*SS2SI, VCVT*SD2SI, VCVT*TSS2SI, and VCVT*TSD2SI instructions, however due to this erratum the VEX.L bit is not ignored and will cause a #UD.

Implication: Unexpected #UDs will be seen when the VEX.L bit is set to one with VCVTSS2SI, VCVT*SD2SI, VCVT*TSS2SI, and VCVT*TSD2SI instructions.

Workaround: Software should ensure that the VEX.L bit is set to 0 for all scalar instructions.

Status: For the steppings affected, see the Table 1.

BDX26 Processor May Livelock During On Demand Clock Modulation

Problem: The processor may livelock when (1) a processor thread has enabled on demand clock modulation via bit 4 of the `IA32_CLOCK_MODULATION MSR` (19AH) and the clock modulation duty cycle is set to 12.5% (02H in bits 3:0 of the same MSR), and (2) the other processor thread does not have on demand clock modulation enabled and that thread is executing a stream of instructions with the lock prefix that either split a cacheline or access UC memory.

Implication: Program execution may stall on both threads of the core subject to this erratum.

Workaround: This erratum will not occur if clock modulation is enabled on all threads when using on demand clock modulation or if the duty cycle programmed in the `IA32_CLOCK_MODULATION MSR` is 18.75% or higher.

Status: For the steppings affected, see the Table 1.

BDX27 Performance Monitor Events OTHER_ASSISTS.AVX_TO_SSE And OTHER_ASSISTS.SSE_TO_AVX May Over Count

Problem: The Performance Monitor events `OTHER_ASSISTS.AVX_TO_SSE` (Event C1H; Umask 08H) and `OTHER_ASSISTS.SSE_TO_AVX` (Event C1H; Umask 10H) incorrectly increment and over count when a Hardware Lock Elision (HLE) abort occurs.

Implication: The Performance Monitor Events `OTHER_ASSISTS.AVX_TO_SSE` And `OTHER_ASSISTS.SSE_TO_AVX` may over count.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX28 Performance Monitor Event DSB2MITE_SWITCHES.COUNT May Over Count

Problem: The Performance Monitor Event DSB2MITE_SWITCHES.COUNT (Event ABH; Umask 01H) should count the number of Decode Stream Buffer (DSB) to Macro Instruction Translation Engine (MITE) switches. Due to this erratum, the DSB2MITE_SWITCHES.COUNT event will count speculative switches and cause the count to be higher than expected.

Implication: The Performance Monitor Event DSB2MITE_SWITCHES.COUNT may report count higher than expected.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX29 Timed MWAIT May Use Deadline of a Previous Execution

Problem: A timed MWAIT instruction specifies a TSC deadline for execution resumption. If a wake event causes execution to resume before the deadline is reached, a subsequent timed MWAIT instruction may incorrectly use the deadline of the previous timed MWAIT when that previous deadline is earlier than the new one.

Implication: A timed MWAIT may end earlier than expected.

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

Status: For the steppings affected, see the Table 1.

BDX30 IA32_VMX_VMCS_ENUM MSR (48AH) Does Not Properly Report The Highest Index Value Used For VMCS Encoding

Problem: IA32_VMX_VMCS_ENUM MSR (48AH) bits 9:1 report the highest index value used for any VMCS encoding. Due to this erratum, the value 21 is returned in bits 9:1 although there is a VMCS field whose encoding uses the index value 23.

Implication: Software that uses the value reported in IA32_VMX_VMCS_ENUM[9:1] to read and write all VMCS fields may omit one field.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX31 Incorrect FROM_IP Value For an RTM Abort in BTM or BTS May Be Observed

Problem: During Restricted Transactional Memory (RTM) operation when branch tracing is enabled using BTM or BTS, the incorrect EIP value (From_IP pointer) may be observed for an RTM abort.

Implication: Due to this erratum, the From_IP pointer may be the same as that of the immediately preceding taken branch.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX32 Locked Load Performance Monitoring Events May Under Count

Problem: The performance monitoring events MEM_TRANS_RETIRED.LOAD_LATENCY (Event CDH; Umask 01H), MEM_LOAD_RETIRED.L2_HIT (Event D1H; Umask 02H), and MEM_UOPS_RETIRED.LOCKED (Event DOH; Umask 20H) should count the number of locked loads. Due to this erratum, these events may under count for locked transactions that hit the L2 cache.

Implication: The above event count will under count on locked loads hitting the L2 cache.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX33 Transactional Abort May Cause an Incorrect Branch Record

Problem: If an Intel® Transactional Synchronization Extensions (Intel® TSX) transactional abort event occurs during a string instruction, the From-IP in the LBR is not correctly reported.

Implication: Due to this erratum, an incorrect FROM-IP on the top of LBR stack may be observed.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX34 PMI May be Signaled More Than Once For Performance Monitor Counter Overflow

Problem: Due to this erratum, Performance Monitoring Interrupt (PMI) may be repeatedly issued until the counter overflow bit is cleared in the overflowing counter.

Implication: Multiple PMIs may be received when a performance monitor counter overflows.

Workaround: None identified. If the PMI is programmed to generate an NMI, software may delay the End-of- Interrupt (EOI) register write for the interrupt until after the overflow indications have been cleared.

Status: For the steppings affected, see the Table 1.

BDX35 Execution of FXSAVE or FXRSTOR With the VEX Prefix May Produce a #NM Exception

Problem: Attempt to use FXSAVE or FXRSTOR with a VEX prefix should produce an Invalid-Opcode (#UD) exception. If either the TS or EM flag bits in CR0 are set, a #NM (device-not-available) exception will be raised instead of #UD exception.

Implication: Due to this erratum, a #NM exception may be signaled instead of a #UD exception on an FXSAVE or an FXRSTOR with a VEX prefix.

Workaround: Software should not use FXSAVE or FXRSTOR with the VEX prefix.

Status: For the steppings affected, see the Table 1.

BDX36 VM Exit May Set IA32_EFER.NXE When IA32_MISC_ENABLE Bit 34 Is Set to 1

Problem: When "XD Bit Disable" in the `IA32_MISC_ENABLE MSR` (1A0H) bit 34 is set to 1, it should not be possible to enable the "execute disable" feature by setting `IA32_EFER.NXE`. Due to this erratum, a VM exit that occurs with the 1-setting of the "load `IA32_EFER`" VM-exit control may set `IA32_EFER.NXE` even if `IA32_MISC_ENABLE` bit 34 is set to 1. This erratum can occur only if `IA32_MISC_ENABLE` bit 34 was set by guest software in VMX non-root operation.

Implication: Software in VMX root operation may execute with the "execute disable" feature enabled despite the fact that the feature should be disabled by the `IA32_MISC_ENABLE MSR`. Intel has not observed this erratum with any commercially available software.

Workaround: A virtual-machine monitor should not allow guest software to write to the `IA32_MISC_ENABLE MSR`.

Status: For the steppings affected, see the Table 1.

BDX37 A MOV to CR3 When EPT Is Enabled May Lead to an Unexpected Page Fault or an Incorrect Page Translation

Problem: If Extended Page Tables (EPT) is enabled, a MOV to CR3 or VMFUNC may be followed by an unexpected page fault or the use of an incorrect page translation.

Implication: Guest software may crash or experience unpredictable behavior as a result of this erratum.

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

Status: For the steppings affected, see the Table 1.

BDX38 Intel® Processor Trace (Intel® PT) Packet Generation May Stop Sooner Than Expected

Problem: Setting the STOP bit (bit 4) in a Table of Physical Addresses entry directs the processor to stop Intel® PT packet generation when the associated output region is filled. The processor indicates this has occurred by setting the Stopped bit (bit 5) of `IA32_RTIT_STATUS MSR` (571H). Due to this erratum, packet generation may stop earlier than expected.

Implication: When this erratum occurs, the OutputOffset field (bits [62:32]) of the `IA32_RTIT_OUTPUT_MASK_PTRS MSR` (561H) holds a value that is less than the size of the output region which triggered the STOP condition; Intel® PT analysis software should not attempt to decode packet data bytes beyond the OutputOffset.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX39 PEBS Eventing IP Field May Be Incorrect After Not-Taken Branch

Problem: When a PEBS record is logged immediately after a not-taken conditional branch (Jcc instruction), the Eventing IP field should contain the address of the first byte of the Jcc instruction. Due to this erratum, it may instead contain the address of the instruction preceding the Jcc instruction.

Implication: Performance monitoring software using PEBS may incorrectly attribute PEBS events that occur on a Jcc to the preceding instruction.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX40 Reading The Memory Destination of an Instruction That Begins an HLE Transaction May Return the Original Value

Problem: An HLE transactional region begins with an instruction with the XACQUIRE prefix. Due to this erratum, reads from within the transactional region of the memory destination of that instruction may return the value that was in memory before the transactional region began.

Implication: Due to this erratum, unpredictable system behavior may occur.

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

Status: For the steppings affected, see the Table 1.

BDX41 Removed Due to Inapplicability

BDX42 Performance Monitoring Event INSTR_RETIRED.ALL May Generate Redundant PEBS Records For an Overflow

Problem: Due to this erratum, the performance monitoring feature PDIR for `INSTR_RETIRED.ALL` (Event C0H; Umask 01H) will generate redundant PEBS records for a counter overflow. This can occur if the lower 6 bits of the performance monitoring counter are not initialized or reset to 0, in the PEBS counter reset field of the DS Buffer Management Area.

Implication: The performance monitor feature PDIR, may generate redundant PEBS records for an overflow.

Workaround: Initialize or reset the counters such that lower 6 bits are 0.

Status: For the steppings affected, see the Table 1.

BDX43 Reset During PECI Transaction May Cause a Machine Check Exception

Problem: If a PECI transaction is interrupted by a warm reset, it may result in a machine check exception with MCACOD of 0x402.

Implication: When this erratum occurs, the system becomes unresponsive and a machine check will be generated.

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

Status: For the steppings affected, see the Table 1.

BDX44 Intel® PT MODE.Exec, PIP, and CBR Packets Are Not Generated as Expected

Problem: The Intel® PT MODE.Exec (MODE packet – Execution mode leaf), PIP, and Core: Bus Ratio (CBR) packets are generated at the following Packet Stream Boundary (PSB+) event rather than at the time of the originating event as expected.

Implication: The decoder may not be able to properly disassemble portions of the binary or interpret portions of the trace because many packets may be generated between the MODE.Exec, PIP, and CBR events and the following PSB+ event.

Workaround: The processor inserts these packets as status packets in the PSB+ block. The decoder may have to skip forward to the next PSB+ block in the trace to obtain the proper updated information to continue decoding.

Status: For the steppings affected, see the Table 1.

BDX45 Performance Monitor Instructions Retired Event May Not Count Consistently

Problem: The Performance Monitor Instructions Retired event (Event C0H; Umask 00H) and the instruction retired fixed counter IA32_FIXED_CTR0 MSR (309H) are used to count the number of instructions retired. Due to this erratum, certain internal conditions may cause the counters to increment when no instruction has retired or to intermittently not increment when instructions have retired.

Implication: A performance counter counting instructions retired may over count or under count. The count may not be consistent between multiple executions of the same code.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX46 General-Purpose Performance Counters May be Inaccurate with Any Thread

Problem: The IA32_PMCx MSR (C1H - C8H) general-purpose performance counters may report inaccurate counts when the associated event selection IA32_PERFEVTSELx MSR's (186H - 18DH) AnyThread field (bit 21) is set and either.

Implication: Due to this erratum, IA32_PMCx counters may be inaccurate.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX47 An Invalid LBR May Be Recorded Following a Transactional Abort

Problem: Use of Intel® TSX may result in a transactional abort. If an abort occurs immediately following a branch instruction, an invalid LBR may be recorded before the LBR produced by the abort.

Implication: The invalid LBR may interfere with execution path reconstruction prior to the transactional abort.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX48 Executing an RSM Instruction With Intel® PT Enabled Will Signal a #GP

Problem: Upon delivery of an SMI, the processor saves and then clears TraceEn in the IA32_RTIT_CTL MSR (570H), thus disabling Intel® PT. If the SMI handler enables Intel® PT and it remains enabled when an RSM instruction is executed, a shutdown event should occur. Due to this erratum, the processor does not shut down but instead generates a #GP.

Implication: When this erratum occurs, a #GP will be signaled.

Workaround: If software enables Intel® PT in system-management mode, it should disable Intel® PT before executing RSM.

Status: For the steppings affected, see the Table 1.

BDX49 Intel PT PIP May be Unexpectedly Generated

Problem: When Intel® PT is enabled, PSB+ packets may include a PIP even though the OS field (bit 2) of IA32_RTIT_CTL MSR (570H) is 0.

Implication: When this erratum occurs, user-mode tracing (indicated by IA32_RTIT_CTL.OS = 0) may include CR3 address information. This may be an undesirable leakage of kernel information.

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

Status: For the steppings affected, see the Table 1.

BDX50 Processor Core Ratio Changes While in Probe Mode May Result in a Hang

Problem: If a processor core ratio change occurs while the processor is in probe mode, the system may hang.

Implication: Due to this erratum, the processor may hang.

Workaround: None identified. Processor core ratio changes may be disabled to avoid this erratum.

Status: For the steppings affected, see the Table 1.

BDX51 Processor Does Not Check IRTE Reserved Bits

Problem: As per the Intel® Virtualization Technology (Intel® VT) for Directed I/O (Intel® VT-d) specification, bits 63:HAW (Host Address Width) of the Posted Interrupt Descriptor Upper Address field in the Interrupt Remapping Table Entry (IRTE) must be checked for a value of 0; violations must be reported as an interrupt-remapping fault. Due to this erratum, hardware does not perform this check and does not signal an interrupt-remapping fault on violations.

Implication: If software improperly programs the reserved address bits of posted interrupt descriptor upper address in the IRTE to a value other than zero, hardware will not detect and report the violation.

Workaround: Software must ensure posted interrupt address bits 63:HAW in the IRTE are zero.

Status: For the steppings affected, see the Table 1.

BDX52 PCIe* TPH Request Capability Structure Incorrectly Advertises Device Specific Mode as Supported

Problem: The Transaction layer packet Processing Hints (TPH) Requester Capability Structure (PCIe* Capability ID type 0017H) incorrectly reports that Device Specific Mode is supported in its TPH Requester Capability Register (bit 2 at offset 04H in the capability structure).

Implication: The processor supports only No Steering Tag (ST) Mode. The PCIe* Base Specification allows, in this instance, the TPH Requester Capability Structure's TPH Requester Control Register (at offset 08H) bits 2:0 to be hardwired to '000', forcing No ST Mode. Advertising Device Specific Mode but forcing No ST Mode is a violation of the PCIe* Base Specification (and may be reported as a compliance issue). Intel has not observed this erratum to impact the operation of any commercially available system.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX53 Package C3 State or Deeper May Lead to a Reset

Problem: Due to this erratum, the processor may reset and signal a Machine Check error with a `IA32_MCi_STATUS.MCACOD` value of 0400H when in Package C3 state or deeper.

Implication: When this erratum occurs, the processor will reset and report an uncorrectable machine check error.

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

Status: For the steppings affected, see the Table 1.

BDX54 VMX-Preemption Timer May Stop Operating When ACC Is Enabled

Problem: When the `MSR_PKG_CST_CONFIG_CONTROL.ACC_Enable` bit (MSR E2H, bit 16) is set, the VMX-preemption timer is not decremented in the HLT state.

Implication: When Autonomous C-State Control (ACC) is enabled, the VMX-preemption timer may not cause a VM exit when expected.

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

Status: For the steppings affected, see the Table 1.

BDX55 Intel® Advanced Vector Extensions (Intel® AVX) Workloads May Exceed ICCMAX Limits

Problem: Intel® AVX workloads require a reduced maximum turbo ratio. Due to this erratum, the Intel® AVX turbo ratio is higher than expected which may cause the processor to exceed ICCMAX limits and lead to unpredictable system behavior.

Implication: Due to this erratum, the processor may exhibit unpredictable system behavior.

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

Status: For the steppings affected, see the Table 1.

BDX56 Writing MSR_ERROR_CONTROL May Cause a #GP

Problem: A WRMSR that attempts to set MODE1_MEMERROR_REPORT field (bit 1) and/or MEM_CORRERR_LOGGING_DISABLE field (bit 5) of the MSR_ERROR_CONTROL MSR (17FH) may incorrectly cause a #GP.

Implication: Due to this erratum, if BIOS attempts to change the value of the listed bits, a #GP may occur.

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

Status: For the steppings affected, see the Table 1.

BDX57 Enabling ACC in VMX Non-Root Operation May Cause System Instability

Problem: ACC is enabled by setting ACC_Enable (bit 16) of MSR_PKG_CST_CONFIG_CONTROL (E2H) to '1'. If ACC is enabled while the processor is in VMX non-root operation, an unexpected VM exit, a machine check, or unpredictable system behavior may result.

Implication: Enabling ACC may lead to system instability.

Workaround: None identified. BIOS should not enable ACC.

Status: For the steppings affected, see the Table 1.

BDX58 A Spurious Patrol Scrub Error May Be Logged

Problem: When a memory ECC error occurs, a spurious patrol scrub error may also be logged on another memory channel.

Implication: A patrol scrub correctable error may be incorrectly logged.

Workaround: The Home Agent error registers and correctable error count registers (Bus 1; Device 20; Function 2; Offset 104-110) provides accurate error information.

Status: For the steppings affected, see the Table 1.

BDX59 Performance Monitoring Counters May Produce Incorrect Results for BR_INST_RETIRED Event on Logical Processor

Problem: Performance monitoring event BR_INST_RETIRED (C4H) counts retired branch instructions. Due to this erratum, when operating on logical processor 1 of any core, BR_INST_RETIRED.FAR_BRANCH (Event C4H; Umask 40H) and BR_INST_RETIRED.ALL_BRANCHES (Event C4H; Umask 04H) may count incorrectly. Logical processor 0 of all cores and cores with SMT disabled are not affected by this erratum.

Implication: Due to this erratum, certain performance monitoring event may produce unreliable results when SMT is enabled.

Workaround: Due to this erratum, certain performance monitoring event may produce unreliable results when SMT is enabled.

Status: For the steppings affected, see the Table 1.

BDX60 An APIC Timer Interrupt During Core C6 Entry May Be Lost

Problem: Due to this erratum, an APIC timer interrupt coincident with the core entering C6 state may be lost rather than held for servicing later.

Implication: A lost APIC timer interrupt may lead to missed deadlines or a system hang.

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

Status: For the steppings affected, see the Table 1.

BDX61 Processor Instability May Occur When Using The PECI RdIAMSR Command

Problem: Under certain circumstances, reading a machine check register using the PECI RdIAMSR command may result in a machine check, processor hang or shutdown.

Implication: Machine check, hang or shutdown may be observed when using the PECI RdIAMSR command.

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

Status: For the steppings affected, see the Table 1.

BDX62 A #VE May Not Invalidate Cached Translation Information

Problem: An Extended Page Table (EPT) violation that causes a Virtualization Exception (#VE) may not invalidate the guest-physical mappings that were used to translate the guest-physical address that caused the EPT violation.

Implication: Due to this erratum, the system may hang.

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

Status: For the steppings affected, see the Table 1.

BDX63 Package C-state Transitions While Inband PECI Accesses Are in Progress May Cause Performance Degradation

Problem: When a Package C-state transition occurs at the same time an inband PECI transaction occurs, PROCHOT# may be incorrectly asserted.

Implication: Incorrect assertion of PROCHOT# reduces the core frequency to the minimum operating frequency of 1.2GHz resulting in persistent performance degradation.

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

Status: For the steppings affected, see the Table 1.

BDX64 Attempting Concurrent Enabling of Intel® PT With LBR, BTS, or BTM Results in a #GP

Problem: If LBR, BTS, or BTM are enabled in the IA32_DEBUGCTL MSR (1D9H), an attempt to enable Intel® PT in `IA32_RTIT_CTL MSR` (570H) results in a #GP. (Notice that the BTM enable bit in `IA32_DEBUGCTL MSR` is named “TR”.) Correspondingly, if Intel® PT was previously enabled when an attempt is made to enable LBR, BTS, or BTM, a #GP will occur.

Implication: An unexpected #GP may occur when concurrently enabling any one of LBR, BTS, or BTM with Intel® PT.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX65 A DDR4 C/A Parity Error in Lockstep Mode May Result in a Spurious Uncorrectable Error

Problem: If a memory Command/Address (C/A) parity error occurs while the memory subsystem is configured in lockstep mode then the channel that observed the error will properly log the error but the associated channel in lockstep will incorrectly log an uncorrectable error in its `IA32_MCi_STATUS MSR`.

Implication: Due to this erratum, incorrect logging of an uncorrectable memory error in `IA32_MCi_STATUS` may occur.

Status: A BIOS code change has been identified and may be implemented as a workaround for this erratumA

BDX66 Cores May be Unable to Reach Maximum Turbo Frequency

Problem: Due to this erratum, processors with more than ten cores may be limited to less than the specified maximum turbo frequency.

Implication: When this erratum occurs, the processor performance is reduced.

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

Status: For the steppings affected, see the Table 1.

BDX67 PEBS Record May Be Generated After Being Disabled

Problem: A performance monitoring counter may generate a PEBS record after disabling PEBS or the performance monitoring counter by clearing the corresponding enable bit in `IA32_PEBS_ENABLE MSR` (3F1H) or `IA32_PERF_GLOBAL_CTRL MSR` (38FH).

Implication: A PEBS record generated after a VMX transition will store into memory according to the post-transition Debug Store (DS) configuration. These stores may be unexpected if PEBS is not enabled following the transition.

Workaround: It is possible for the BIOS to contain a workaround for this erratum. A software workaround is possible through disallowing PEBS during VMX non-root operation and disabling PEBS prior to VM entry.

Status: For the steppings affected, see the Table 1.

BDX68 Removed Due to Duplication of BDX76

BDX69 Removed Due to Inapplicability

BDX70 Removed Due to Duplication of BDX72

BDX71 PEBS EventingIP Field May Be Incorrect Under Certain Conditions

Problem: The EventingIP field in the PEBS record reports the address of the instruction that triggered the PEBS event. Under certain complex microarchitectural conditions, the EventingIP field may be incorrect.

Implication: When this erratum occurs, performance monitoring software may not attribute the PEBS events to the correct instruction.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX72 Turbo May Be Delayed After Exiting C6 When Using HWP

Problem: Due to this erratum, enabling Hardware-Controlled Performance States (HWP) by setting bit 0 of IA32_PM_ENABLE (MSR 770H) may lead to an unexpected delay in reaching turbo frequencies after a core exits C6 sleep state. This erratum does not occur when HWP is not enabled.

Implication: When this erratum occurs, enabling HWP may lead to a visible reduction of system performance.

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

Status: For the steppings affected, see the Table 1.

BDX73 Writing The IIO_LLC_WAYS MSR Results in an Incorrect Value

Problem: Writing the IIO_LLC_WAYS MSR (C8Bh) always sets bits [1:0] regardless of the value written.

Implication: IIO cache way allocation may not act as intended. Intel has not seen any functional failure due to this erratum.

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

Status: For the steppings affected, see the Table 1.

BDX74 RF May Be Incorrectly Set in the EFLAGS That Is Saved on a Fault in PEBS or BTS

Problem: After a fault due to a failed PEBS or BTS address translation, the Resume Flag (RF) may be incorrectly set in the EFLAGS image that is saved.

Implication: When this erratum occurs, a code breakpoint on the instruction following the return from handling the fault will not be detected. This erratum only happens when the user does not prevent faults on PEBS or BTS.

Workaround: Software should always prevent faults on PEBS or BTS.

Status: For the steppings affected, see the Table 1.

BDX75 The System May Hang When Executing a Complex Sequence of Locked Instructions

Problem: Under certain internal timing conditions while executing a complex sequence of locked instructions, the system may hang.

Implication: The system may hang while executing a complex sequence of locked instructions and cause an Internal Timeout Error Machine Check (IA32_MCi_STATUS.MCACOD=0400H).

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

Status: For the steppings affected, see the Table 1.

BDX76 Using Intel® TSX Instructions May Lead to Unpredictable System Behavior

Problem: Under complex microarchitectural conditions, software using Intel® TSX may result in unpredictable system behavior. Intel has only seen this under synthetic testing conditions. Intel is not aware of any commercially available software exhibiting this behavior.

Implication: Due to this erratum, unpredictable system behavior may occur.

Workaround: It is possible for BIOS to contain a workaround for this erratum.*Performance Monitoring Impact of Intel® Transactional Synchronization Extension Memory Ordering Issue*

Status: For the steppings affected, see the Table 1.

BDX77 Data Breakpoint Coincident With a Machine Check Exception May be Lost

Problem: If a data breakpoint occurs coincident with a machine check exception, then the data breakpoint may be lost.

Implication: Due to this erratum, a valid data breakpoint may be lost.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX78 Internal Parity Errors May Incorrectly Report Overflow in the IA32_MC0_STATUS MSR

Problem: Due to this erratum, an uncorrectable internal parity error with an `IA32_MC0_STATUS.MCACOD` (bits [15:0]) value of 0005H may incorrectly set the `IA32_MC0_STATUS.OVER` flag (bit 62) indicating an overflow when a single error has been observed.

Implication: `IA32_MC0_STATUS.OVER` may not accurately indicate multiple occurrences of errors. There is no other impact to normal processor functionality.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX79 Incorrect VMCS Used for PML-Index field on VMX Transitions Into and Out of SMM

Problem: The Page Modification Log (PML) index field is saved to an incorrect VMCS on an SMM VM exit. VM entries that return from SMM restore the PML-index field from that same incorrect VMCS.

Implication: The PML-index field is correctly maintained for expected use cases, in which the SMM-Transfer Monitor (STM) does not access the PML-index field in the SMM VMCS. If the STM uses VMREAD to read the field, it will get an incorrect value. In addition, the processor will ignore any modification of the field that the STM makes using VMWRITE. Intel has not observed this erratum to impact any commercially available software.

Workaround: None identified. To access the PML-index field, STM software should first load the current-VMCS pointer with a pointer to the executive VMCS.

Status: For the steppings affected, see the Table 1.

BDX80 An APIC Timer Interrupt During Core C6 Entry May Be Lost

Problem: Due to this erratum, an APIC timer interrupt coincident with the core entering C6 state may be lost rather than held for servicing later.

Implication: A lost APIC timer interrupt may lead to missed deadlines or a system hang.

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

Status: For the steppings affected, see the Table 1.

BDX81 Inband PECI Concurrent With OS Patch Load May Result in Incorrect Throttling Causing Reduced System Performance

Problem: Microcode updates loaded by the operating system may result in excessive and persistent throttling that significantly reduces system performance.

Implication: When this erratum occurs, performance may be reduced, concurrent with an incorrect assertion of the PROCHOT# signal.

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

Status: For the steppings affected, see the Table 1.

BDX82 An Intel® Hyper-Threading Technology (Intel® HT Technology) Enabled Processor May Exhibit Internal Parity Errors or Unpredictable System Behavior

Problem: Under a complex series of micro-architectural events while running Intel® Hyper-Threading Technology, a correctable internal parity error or unpredictable system behavior may occur.

Implication: A correctable error (IA32_MC0_STATUS.MCACOD=0005H and IA32_MC0_STATUS.MSCOD=0001H) may be logged. The unpredictable system behavior frequently leads to faults (for example, #UD, #PF, #GP).

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

Status: For the steppings affected, see the Table 1.

BDX83 IA32_MC4_STATUS.VAL May be Incorrectly Cleared by Warm Reset

Problem: Due to this erratum, the IA32_MC4_STATUS. VAL (MSR 411H, bit 63) may be incorrectly cleared by a warm reset.

Implication: Software may be unaware that a machine check occurred before the warm reset.

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

Status: For the steppings affected, see the Table 1.

BDX84 Some DRAM And L3 Cache Performance Monitoring Events May Undercount

Problem: Due to this erratum, the supplier may be misattributed to unknown, and the following events may undercount:

```
MEM_LOAD_UOPS_RETIRED.L3_HIT (Event D1H Umask 04H)
MEM_LOAD_UOPS_RETIRED.L3_MISS (Event D1H Umask 20H)
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_MISS (Event D2H Umask 01H)
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HIT (Event D2H Umask 02H)
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_HITM (Event D2H Umask 04H)
MEM_LOAD_UOPS_L3_HIT_RETIRED.XSNP_NONE (Event D2H Umask 08H)
MEM_LOAD_UOPS_L3_MISS_RETIRED.LOCAL_DRAM (Event D3H Umask 01H)
MEM_TRANS_RETIRED.LOAD_LATENCY (Event CDH Umask 01H)
```

Implication: The affected events may undercount, resulting in inaccurate memory profiles. For the affected events that are precise, PEBS records may be generated at incorrect points. Intel has observed incorrect counts by as much as 20%.

Workaround: None Identified.

Status: For the steppings affected, see the Table 1.

BDX85 An x87 Store Instruction Which Pends #PE While EPT Is Enabled May Lead to an Unexpected Machine Check and/or Incorrect x87 State Information

Problem: The execution of an x87 store instruction which causes a Precision Exception (#PE) to be pended and also causes a VM-exit due to an EPT violation or misconfiguration may lead the VMM logging a machine check exception with a cache hierarchy error (IA32_MCi_STATUS.MCACOD = 0150H and IA32_MCi_STATUS.MSCOD = 000FH). Additionally, FSW.PE and FSW.ES (bits 5 and 7 of the FPU Status Word) may be incorrectly set to 1, and the x87 Last Instruction Opcode (FOP) may be incorrect.

Implication: When this erratum occurs, the VMM may receive an expected machine check exception and software attempting to handle the #PE may not behave as expected.

Workaround: None Identified.

Status: For the steppings affected, see the Table 1.

BDX86 Load Latency Performance Monitoring Facility May Stop Counting

Problem: The performance monitoring events MEM_TRANS_RETIRED.LOAD_LATENCY_* (Event CDH; UMask 01H; any latency) count load instructions whose latency exceed a predefined threshold, where the loads are randomly selected using the Load Latency facility (PEBS extension). However, due to this erratum, load latency facility may stop counting load instructions when Intel® HT Technology is enabled.

Implication: Counters programmed with the affected events stop incrementing and do not generate PEBS records.

Workaround: None Identified.

Status: For the steppings affected, see the Table 1.

BDX87 General-Purpose Performance Monitoring Counters 4-7 Will Not Increment Do Not Count With USR Mode Only Filtering

Problem: The IA32_PMC4-7 MSR (C5H-C8H) general-purpose performance monitoring counters will not count when the associated CPL filter selection in IA32_PERFEVTSELx MSR's (18AH-18DH) USR field (bit 16) is set while OS field (bit 17) is not set.

Implication: Software depending upon IA32_PMC4-7 to count only USR events will not operate as expected. Counting OS only events or OS and USR events together is unaffected by this erratum.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX88 Writing MSR_LASTBRANCH_x_FROM_IP and MSR_LER_FROM_LIP May #GP When Intel® TSX Is Not Supported

Problem: Due to this erratum, on processors that do not support Intel® TSX (CPUID.07H.EBX bits 4 and 11 are both zero), writes to MSR_LASTBRANCH_x_FROM_IP (MSR 680H to 68FH) and MSR_LER_FROM_LIP (MSR 1DDH) may #GP unless bits[62:61] are equal to bit[47].

Implication: The value read from MSR_LASTBRANCH_x_FROM_IP and MSR_LER_FROM_LIP is unaffected by this erratum; bits [62:61] contain IN_TSX and TSX_ABORT information respectively. Software restoring these MSRs from saved values are subject to this erratum.

Workaround: Before writing MSR_LASTBRANCH_x_FROM_IP and MSR_LER_FROM_LIP, ensure the value being written has bit[47] replicated in bits[62:61]. This is most easily accomplished by sign extending from bit[47] to bits[62:48].

Status: For the steppings affected, see the Table 1.

BDX89 APIC Timer Interrupt May Not be Generated at the Correct Time In TSC-Deadline Mode

Problem: After writing to the IA32_TSC_ADJUST MSR (3BH), any subsequent write to the IA32_TSC_DEADLINE MSR (6E0H) may incorrectly process the desired deadline. When this erratum occurs, the resulting timer interrupt may be generated at the incorrect time.

Implication: When the local APIC timer is configured for TSC-Deadline mode, a timer interrupt may be generated much earlier than expected or much later than expected. Intel has not observed this erratum with most commercially available software.

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

Status: For the steppings affected, see the Table 1.

BDX90 Loading Microcode Updates or Executing an Authenticated Code Module May Result in a System Hang

Problem: An uncorrectable error (`IA32_MC3_STATUS.MCACOD=0400` and `IA32_MC3_STATUS.MSCOD=0080`) may be logged for processors that have more than 2.5MB last-level-cache per core on attempting to load a microcode update or execute an authenticated code module. This issue does not occur with microcode updates with a signature of 0x0b000021 and greater.

Implication: Due to this erratum, the processor may hang when attempting to load a microcode update or execute an authenticated code module.

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

Status: For the steppings affected, see the Table 1.

BDX91 NVDIMM Data May Not be Preserved Correctly on Power Loss or ADR Activation

Problem: When entering Asynchronous DRAM Self-Refresh (ADR), whether through power loss or a specific ADR command, concurrent reads to the NVDIMM may prevent the data from being properly preserved.

Implication: After an ADR event, memory data may be incorrect and may lead to an ECC error on next access.

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

Status: For the steppings affected, see the Table 1.

BDX92 Link Down Events Behind PCIe* Device Connected to CPU Root Ports Can Cause CTO > 50ms on Other Root Ports

Problem: When a downstream switch connected to a CPU Root Port experiences a link down it may cause a back pressure event that prevents other CPU root ports from completing transaction for >50ms but less than 100ms.

Implication: When intentionally disabling a PCIe* link in the system the IIO Arbiter can get stuck for > 50ms causing other endpoints to exceed their CT value (of 50ms) which is reported as a fatal system ERR2 condition.

Workaround: Set PCIe* CTOs to 100ms or greater if in a vulnerable configuration.

Status: For the steppings affected, see the Table 1.

BDX93 Reads From MSR_LER_TO_LIP May Not Return a Canonical Address

Problem: Due to this erratum, reads from `MSR_LER_TO_LIP` (MSR 1DEH) may return values for bits[63:61] that are not equal to bit[47].

Implication: Reads from `MSR_LER_TO_LIP` may return a non-canonical address where bits[63:61] may be incorrect. Using this value as an address, including restoring the MSR value that was read, may cause a #GP.

Workaround: Software should ensure the value read in `MSR_LER_TO_LIP` bit[47] is replicated in bits[63:61]. This is most easily accomplished by sign extending from bit[47] to bits[63:48].

Status: For the steppings affected, see the Table 1.

BDX94 Processor May Hang After Multiple Microcode Updates Loaded

Problem: Under certain conditions, a microcode update load may hang if another microcode update was already loaded, resulting in an Internal Timer Error Machine Check (`IA32_MCi_STATUS.MCACOD=400H`; bits 15:0).

Implication: Due to this erratum, the processor may hang.

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

Status: For the steppings affected, see the Table 1.

BDX95 In eMCA2 Mode, When the Retirement Watchdog Timeout Occurs CATERR# May be Asserted

Problem: A Retirement Watchdog Timeout (MCACOD = 0x0400) in Enhanced MCA2 (eMCA2) mode will cause the CATERR# pin to be pulsed in addition to an MSMI# pin assertion. In addition, a Machine Check Abort (#MC) will be pended in the cores along with the MSMI.

Implication: Due to this erratum, systems that expect to only see MSMI# will also see CATERR# pulse when a Retirement Watchdog Timeout occurs. The CATERR# pulse can be safely ignored.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX96 Systems That Enable Both OSB and IODC May Exhibit Unexpected System Behavior

Problem: If a platform with four or more sockets is configured to enable both Opportunistic Snoop Broadcast (OSB) and Input Output Directory Cache (IODC) or if a platform with two or more sockets is configured to enable OSB, IODC, and Cluster on Die (CoD), then the system may exhibit unexpected behavior.

Implication: Due to this erratum, the system may exhibit unexpected system behavior.

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

Status: For the steppings affected, see the Table 1.

BDX97 MBM Counters May Report System Memory Bandwidth Incorrectly

Problem: Memory Bandwidth Monitoring (MBM) counters track metrics according to the assigned Resource Monitor ID (RMID) for that logical core. The `IA32_QM_CTR` register (MSR 0xC8E), used to report these metrics, may report incorrect system bandwidth for certain RMID values.

Implication: Due to this erratum, system memory bandwidth may not match what is reported.

Workaround: It is possible for software to contain code changes to work around this erratum. See the white paper titled Intel® Resource Director Technology (Intel® RDT) Reference Manual found at https://software.intel.com/en-us/intel-resource-director-technology-rdt-reference-manual for more information.

Status: For the steppings affected, see the Table 1.

BDX98 When Operating at Maximum Turbo Frequencies, The Processor May Hang

Problem: Under rare microarchitectural conditions, if the processor is operating at maximum turbo frequencies, it may hang without logging any Machine Check Exceptions (MCEs) or other Internal Errors (IERRs).

Implication: When this erratum occurs, the system will hang with no active machine check exceptions or other internal errors.

Workaround: It is possible for software to limit the maximum ratios at which the processor may operate.

Status: For the steppings affected, see the Table 1.

BDX99 A Pending Fixed Interrupt May Be Dispatched Before an Interrupt of The Same Priority Completes

Problem: Resuming from C6 Sleep-State, with Fixed Interrupts of the same priority queued (in the corresponding bits of the IRR and ISR APIC registers), the processor may dispatch the second interrupt (from the IRR bit) before the first interrupt has completed and written to the EOI register, causing the first interrupt to never complete.

Implication: Due to this erratum, Software may behave unexpectedly when an earlier call to an Interrupt Handler routine is overridden with another call (to the same Interrupt Handler) instead of completing its execution.

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

Status: For the steppings affected, see the Table 1.

BDX100 Instruction Fetch May Cause Machine Check if Page Size Was Changed Without Invalidation

Problem: This erratum may cause a machine-check error (IA32_MCi_STATUS.MCACOD=005H with IA32_MCi_STATUS.MSCOD=00FH or IA32_MCi_STATUS.MCACOD=0150H with IA32_MCi_STATUS.MSCOD=00FH) on the fetch of an instruction. It applies only if (1) instruction bytes are fetched from a linear address translated using a 4-Kbyte page and cached in the processor; (2) the paging structures are later modified so that these bytes are translated using a large page (2-Mbyte, 4-Mbyte or 1-GByte) with a different physical address (PA), memory type (PWT, PCD and PAT bits), or User/Supervisor (U/S) bit; and (3) the same instruction is fetched after the paging structure modification but before software invalidates any TLB entries for the linear region.

Implication: Due to this erratum, an unexpected machine check with error code 0150H with MSCOD 00FH may occur, possibly resulting in a shutdown. This erratum could also lead to unexpected correctable machine check (IA32_MCi_STATUS.UC=0) with error code 005H with MSCOD 00FH.

Workaround: Software should not write to a paging-structure entry in a way that would change the page size and either the physical address, memory type or User/Supervisor bit. It can instead use one of the following algorithms: first clear the P flag in the relevant paging-structure entry (for example, PDE); then invalidate any translations for the affected linear addresses; and then modify the relevant paging-structure entry to set the P flag and establish the new page size. An alternative algorithm: first change the physical page attributes (combination of physical address, memory type and User/Supervisor bit) in all 4K pages in the affected linear addresses; then invalidate any translations for the affected linear addresses; and then modify the relevant paging-structure entry to establish the new page size.

Status: For the steppings affected, see the Table 1.

BDX101 Overflow Flag in IA32_MC0_STATUS MSR May Be Incorrectly Set

Problem: Under complex micro-architectural conditions, a single internal parity error seen in IA32_MC0_STATUS MSR (401h) with MCACOD (bits 15:0) value of 5h and MSCOD (bits 31:16) value of 7h, may set the overflow flag (bit 62) in the same MSR.

Implication: Due to this erratum, the IA32_MC0_STATUS overflow flag may be set after a single parity error. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDX102 PECI RdIAMSR() to Non-existent MSR Might Result in IERR

Problem: If a PECI RdIAMSR() transaction is initiated to a non-existent MSR before any microcode patch is loaded (either via FIT or by BIOS), the system may signal IERR# with bank 4 logging a MCCOD of 0x0403 and a MSEC_FW code of 0x09.

Implication: Due to this erratum, PECI accesses to unimplemented MSRs may result in IERR# assertion.

Workaround: Avoid RdIAMSR() access to unimplemented MSRs or delay all PECI transactions until 500ms after RESET_N de-assertion.

Status: For the steppings affected, see the Table 1.

BDX103 Retried PECI PCIConfigLocal Register Accesses May Not Operate Correctly

Problem: When the processor requests a PECI PCIConfigLocal read or write command to be retried, and the PECI host immediately retries the command (within 150 us), the processor may fail to correctly process the retried PECI command.

Implication: Due to this erratum, the PECI PCIConfigLocal read command may return incorrect data, and the PECI PCIConfigLocal write command may incorrectly update the target.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

§Extended Data Register (EDX)Extended Accumulator Register (EAX)

Table 1. Intel® Xeon® Processor E7-8800/4800 Integrated Core/Uncore Errata

#	Stepping	Status	Errata
	B0		
BDEX1	X	No Fix	PCIe* UR and CA Responses May be Sent Before Link Enters LER State
BDEX2	X	No Fix	DDR4 CAP Unable to Determine which DIMM Caused CAP
BDEX3	X	No Fix	Error Source ID Not Logged When Performing MMIO Write to Region Outside Memory Endpoint BARs of Downstream Device
BDEX4	X	No Fix	DIMMTEMPSTAT_[2-0] and ALLDIMMTEMPSTAT and PECI Service 14\22 May Return Data for Opposite Intel® C102/C104 and C112/C114 Scalable Memory Buffer channel
BDEX5	X	No Fix	Clear of One dimmtempstat_[2-0] and mxbtempstat CSR ev_asrt_temp* Field Clears All ev_asrt_temp* Fields
BDEX6	X	No Fix	Back-to-Back Page Walks Due to Instruction Fetches May Cause a System Hang
BDEX7	X	No Fix	MOVNTDQA From WC Memory May Pass Earlier Locked Instructions
BDEX8	X	No Fix	JTAG Boundary Scan For Intel® QuickPath Interconnect (Intel® QPI) and PCIe* Lanes May Report Incorrect Stuck at 1 Errors
BDEX9	X	No Fix	Debug Exceptions May Be Lost in The Case Of Machine Check Exception
BDEX10	X	No Fix	Systems With a Node Controller May Encounter a Memory Ordering Issue in OS Kernel
BDEX11	X	No Fix	Enabling Cluster on Die May Lead to Unpredictable System Behavior

BDEX1 PCIe* UR and CA Responses May be Sent Before Link Enters LER State

Problem: Completions with 9 Uncorrectable Response (9UR) and Completer Abort (CA) status should trigger LER. Further, these packets should be dropped upon entering LER. Due to this erratum, these completions may not be dropped when LER is triggered.

Implication: Since these packets contain no data, there is no loss of error containment. These packets will trigger LER mode; the link will be disabled.

Workaround: None.

Status: For the steppings affected, see the Table 1

BDEX2 DDR4 CAP Unable to Determine which DIMM Caused CAP

Problem: Due to a logic issue, DDR4 Command Address Parity (CAP) feature can correct a CAP error, but will not be able to determine the DIMM which failed.(Formally EX2, EX24 and EX28).

Implication: Unable to determine which DIMM had a DDR4 CAP error if more than 1 DIMM Per Channel (DPC).

Workaround: None.

Status: For the steppings affected, see the Table 1.

BDEX3 Error Source ID Not Logged When Performing MMIO Write to Region Outside Memory Endpoint BARs of Downstream Device

Problem: The Root Port does not log the Error Source ID when performing an aligned write to MMIO address within Root Port Aperture but outside the BARs of downstream device.

Implication: Determination of error source not possible on a MMIO write outside BARs of downstream device by reading ErrSrcId register. Only applies to an MMIO write, a read will log error source ID.

Workaround: None.

Status: For the steppings affected, see the Table 1.

BDEX4 DIMMTEMPSTAT_[2-0] and ALLDIMMTEMPSTAT and PECI Service 14\22 May Return Data for Opposite Intel® C102/C104 and C112/C114 Scalable Memory Buffer channel

Problem: In some cases `DIMMTEMPSTAT_[2-0]` and ALLDIMMTEMPSTAT and PECI service 14\22 may return data for opposite Intel® C102/C104 & C112/C114 Scalable Memory Buffer channel. This only applies to channel 0 and 1 of the respective IMC, and channel 2 and 3 are not impacted by this issue.

Implication: SW or FW polling an Intel® C102/C104 & C112/C114 Scalable Memory Buffer DDR channel will not know specific channel the reading is for.

Workaround: SW or FW polling `DIMMTEMPSTAT_[2-0]` and ALLDIMMTEMPSTAT should take this behavior into account.

Status: For the steppings affected, see the Table 1.

BDEX5 Clear of One dimmtempstat_[2-0] and mxbtempstat CSR ev_asrt_temp* Field Clears All ev_asrt_temp* Fields

Problem: A clear (write of 1 to this RW1C field) to a respective `dimmtempstat_[2-0]` and mxbtempstat CSR `ev_asrt_temp*` field clears all `ev_asrt_temp*` fields in the respective CSR.

Implication: Implication: `ev_asrt_temp*` fields are asserted only once on a thermal transition crossing the respective `ev_asrt_temp*` threshold. It is possible that a thermal transition could occur between the reading and clearing `dimmtempstat_[2-0]` and mxbtempstat `ev_asrt_temp*` fields and such transition could be lost (for example: firmware polls `ev_asrt_templo` then clears `ev_asrt_templo`, but in between the reading and the clearing `ev_asrt_temmid` is logged and cleared).

Workaround: Clear `ev_asrt_temp*` fields as soon as possible after reading them. Log all bits on a read of the `ev_asrt_temp*` fields.

Status: For the steppings affected, see the Table 1.

BDEX6 Back-to-Back Page Walks Due to Instruction Fetches May Cause a System Hang

Problem: Multiple code fetches in quick succession that generate page walks may result in a system hang causing an Internal Timer Error (an MCACOD value of 0400H) logged into `IA32_MCi_STATUS` bits [15:0].

Implication: Due to this erratum, the processor may hang and report a machine check.

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

Status: For the steppings affected, see the Table 1.

BDEX7 MOVNTDQA From WC Memory May Pass Earlier Locked Instructions

Problem: An execution of (V)MOVNTDQA (streaming load instruction) that loads from WC (write combining) memory may appear to pass an earlier locked instruction that accesses a different cache line.

Implication: Software that expects a lock to fence subsequent (V)MOVNTDQA instructions may not operate properly.

Workaround: None identified. Software that relies on a locked instruction to fence subsequent executions of (V)MOVNTDQA should insert an MFENCE instruction between the locked instruction and subsequent (V)MOVNTDQA instruction.

Status: For the steppings affected, see the Table 1.

BDEX8 JTAG Boundary Scan For Intel® QuickPath Interconnect (Intel® QPI) and PCIe* Lanes May Report Incorrect Stuck at 1 Errors

Problem: Boundary Scan testing of the Intel® QPI and PCIe* interfaces may incorrectly report a recurring stuck at 1 failure on Intel® QPI and PCIe* receiver lanes. This erratum only affects Boundary Scan testing and does not affect functional operation of the Intel® QPI and PCIe* interfaces.

Implication: This erratum may result in Boundary Scan test failures reported on one or more of the Intel® QPI and PCIe* lanes.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDEX9 Debug Exceptions May Be Lost in The Case Of Machine Check Exception

Problem: If both a machine check exception and a debug exception are pending on the same instruction boundary, then the machine check exception gets priority and the debug exception may be lost, even if the Processor Context Corrupted (PCC) field is cleared in all of the machine check banks (bit 57=0 in all IA32_MCi_STATUS MSR). This can happen in the case that an instruction triggered a data breakpoint while an unrelated machine check event was received.

Implication: Debugging software may fail to operate as expected if a debug exception is lost.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDEX10 Systems With a Node Controller May Encounter a Memory Ordering Issue in OS Kernel

Problem: The processors in a scalable system with a node controller may observe a memory ordering issue in the processor core when executing locked memory instructions in an OS kernel.

Implication: When this erratum occurs, it may result in unpredictable system behavior including system hangs.

Workaround: None identified.

Status: For the steppings affected, see the Table 1.

BDEX11 Enabling Cluster on Die May Lead to Unpredictable System Behavior

Problem: With Cluster on Die (COD) enabled under complex IO/CPU memory accesses utilizing IO Directory Cache (IODC), it is possible for stale data to be read that may cause unpredictable system behavior.

Implication: Due to this erratum, unpredictable system behavior may occur.

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 Table 1.

§

Specification Changes

The Specification Changes listed in this section apply to the following documents:

- *Intel® Xeon® Processor Datasheet, Volume 1 and 2*
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1: Basic Architecture*
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A: Instruction Set Reference Manual A-M*
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2B: Instruction Set Reference Manual N-Z*
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3A: System Programming Guide*
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3B: System Programming Guide*

There are no new Specification Changes in this Specification Update revision.

§

Documentation Changes

There are no Documentation Changes in this Specification Update revision.

§