Intel® Core™ X-Series Processor Family

Specification Update

February 2020

Revision 009

Document Number: 335901-009

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.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. **No computer system can be absolutely secure.** Check with your system manufacturer or retailer or learn more at **intel.com.**

Intel technologies may require enabled hardware, specific software, or services activation. Check with your system manufacturer or retailer.

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.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps

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.

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.

Contents

Table

Revision History

Version	Description	Date
001	Initial release.	May 2017
002	Added Errata SKZ26 to SKZ59	January 2018
003	Added Errata SKZ60 to SKZ64	March 2018
004	Added Errata SKZ65 to SKZ67	July 2018
005	Updated Errata SKZ49, SKZ65 Added Errata SKZ68 to SKZ72	October 2018
006	Added Errata SKZ73 to SKZ76	November 2018
007	Added Errata SKZ77 to SKZ82	February 2019
008	Added Errata SKZ83 to SKZ98	December 2019
009	Added Errata SKZ99	February 2020

§ §

Preface

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

Information types defined in Nomenclature are consolidated into the specification update and are no longer published in other documents.

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

Affected Documents

Document Title	Document Number
Intel® Core™ X-Processor Family Datasheet Volume 1 of 2	335899
Intel® Core™ X-Processor Family Datasheet Volume 2 of 2	335900

Related Documents

Document Title	Document Number/ Location
Intel® 64 and IA-32 Architecture Software Developer's Manual • Volume 1: Basic Architecture • Volume 2A: Instruction Set Reference Manual A-M • Volume 2B: Instruction Set Reference Manual N-Z • Volume 3A: System Programming Guide • Volume 3B: System Programming Guide • A-32 Intel® Architecture Optimization Reference Manual	http://www.intel.com/ products/processor/ manuals/index.htm

Nomenclature

Errata are design defects or errors. These may cause the processor behavior to deviate from published specifications. Hardware and software designed to be used with any given stepping must assume that all errata documented for that stepping are present on all devices.

Specification Changes are modifications to the current published specifications. These changes will be incorporated in 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: Errata remain in the specification update throughout the product's lifecycle, or until a particular stepping is no longer commercially available. Under these circumstances, errata removed from the specification update are archived and available upon request. Specification changes, specification clarifications and documentation changes are removed from the specification update when the appropriate changes are made to the appropriate product specification or user documentation (datasheets, manuals, and so forth).

Identification Information

Component Identification via Programming Interface

The processor 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	15:13	12	11:8	7:4	3:0
	00000000b	0101b		0b	0110b	0101b	varies per 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 EDX register after RESET, Bits [11:8] of the EAX register after the 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.

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. Note that the EDX processor signature value after reset is equivalent to the processor signature output value in the EAX register.

Cache and TLB descriptor parameters are provided in the EAX, EBX, ECX and EDX registers after the CPUID instruction is executed with a 2 in the EAX register.

Summary Tables of Changes

The following tables indicate the errata, specification changes, specification clarifications, or documentation changes which apply to the Product Name of the 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 uses the following notations:

Codes Used in Summary Tables

Stepping

X: Errata exists 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. Errata Summary Table (Sheet 1 of 4)

Number	Steppings		Status	Errata
	U-0	M-0		
SKZ1	X	X	No Fix	A CAP Error While Entering Package C6 May Cause DRAM to Fail to Enter Self-Refresh
SKZ2	X	X	No Fix	PCIe* Lane Error Status Register May Log False Correctable Error
SKZ3	X	X	No Fix	In Memory Mirror Mode, DataErrorChunk Field May be Incorrect
SKZ4	X	X	No Fix	Intel® RDT MBM Does Not Accurately Track Write Bandwidth
SKZ5	X	X	No Fix	PCIe* Port May Incorrectly Log Malformed_TLP Error
SKZ6	X	X	No Fix	Short Loops Which Use AH/BH/CH/DH Registers May Cause Unpredictable System Behavior
SKZ7	X	X	No Fix	Credits Not Returned For PCIe* Packets That Fail ECRC Check
SKZ8	X	X	No Fix	Link Training Error Due to Single Polarity of a PCIe* Differential Data Pair Being Disconnected
SKZ9	X	X	No Fix	IODC Entry 0 Cannot be Masked

Table 1. Errata Summary Table (Sheet 2 of 4)

Number	Steppings		Status	Errata
	U-0	M-0		
SKZ10	X	X	No Fix	With eMCA2 Enabled a 3-Strike May Cause an Unnecessary CATERR# Instead of Only MSMI
SKZ11	X	X	No Fix	CMCI May not be Signaled for Corrected Error
SKZ12	X	X	No Fix	CSRs SVID and SDID are not Implemented For Some DDRIO And PCU Devices
SKZ13	X	X	No Fix	Register Broadcast Read From DDRIO May Return a Zero Value
SKZ14	X	X	No Fix	Intel® CMT Counters May Not Count Accurately
SKZ15	X	X	No Fix	Intel® CAT May Not Restrict Cacheline Allocation Under Certain Conditions
SKZ16	X	X	No Fix	Intel® PCIe* Corrected Error Threshold Does Not Consider Overflow Count When Incrementing Error Counter
SKZ17	X	X	No Fix	IIO RAS VPP Hangs During The Warm Reset Test
SKZ18	X	X	No Fix	For the Steppings affected, refer the Summary Tables of Changes.Processor May Hang on Complex Sequence of Conditions
SKZ19	X	X	No Fix	Intel® PCIe* Root Port Electromechanical Interlock Control Register Can be Written
SKZ20	X	X	No Fix	System Hangs May Occur When IPQ And IRQ Requests Happen at the Same Time
SKZ21	X	X	No Fix	Masked Bytes in a Vector Masked Store Instructions May Cause Write Back of a Cache Line
SKZ22	X	X	No Fix	ERROR_N[2:0] Pins May not be Cleared After a Warm Reset
SKZ23	X	X	No Fix	Intel® PCIe* Slot Presence Detect and Presence Detect Changed Logic not PCIe* Specification Compliant
SKZ24	X	X	No Fix	Debug Exceptions May be Lost or Misreported When MOV SS or POP SS Instruction is not Followed by a Write to SP
SKZ25	X	X	No Fix	Incorrect Branch Predicted Bit in BTS/BTM Branch Records
SKZ26	X	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
SKZ27	X	X	No Fix	Intel® PT TIP.PGD May Not Have Target IP Payload
SKZ28	X	X	No Fix	The Corrected Error Count Overflow Bit in IA32_ MC0_STATUS is not Updated When the UC Bit is Set
SKZ29	X	X	No Fix	SMRAM State-Save Area Above the 4GB Boundary May Cause Unpredictable System Behavior
SKZ30	X	X	No Fix	VM Exit May Set IA32_EFER.NXE When IA32_MISC_ENABLE Bit 34 is Set to 1
SKZ31	X	X	No Fix	x87 FPU Exception (#MF) May be Signaled Earlier than Expected
SKZ32	X	X	No Fix	POPCNT Instruction May Take Longer to Execute than Expected
SKZ33	X	X	No Fix	Load Latency Performance Monitoring Facility May Stop Counting
SKZ34	X	X	No Fix	Intel® Processor Trace PSB+ Packets May Contain Unexpected Packets
SKZ35	X	X	No Fix	Performance Monitoring Counters May Undercount When Using CPL Filtering
SKZ36	X	X	No Fix	Intel® PT ToPA PMI Does not Freeze Performance Monitoring Counters
SKZ37	X	X	No Fix	Performance Monitoring Load Latency Events May be Inaccurate For Gather Instructions
SKZ38	X	X	No Fix	CPUID TLB Associativity Information is Inaccurate
SKZ39	X	X	No Fix	Vector Masked Store Instructions May Cause Write Back of Cache Line Where Bytes are Masked
SKZ40	X	X	No Fix	Incorrect FROM_IP Value For an RTM Abort in BTM or BTS May be Observed
SKZ41	X	X	No Fix	PEBS Record After a WRMSR to IA32_BIOS_UPDT_TRIG May be Incorrect
SKZ42	X	X	No Fix	MOVNTDQA From WC Memory May Pass Earlier Locked Instructions
SKZ43	X	X	No Fix	#GP on Segment Selector Descriptor that Straddles Canonical Boundary May Not Provide Correct Exception Error Code

Table 1. Errata Summary Table (Sheet 3 of 4)

Number	Steppings		Status	Errata
	U-0	M-0		
SKZ44	X	X	No Fix	Intel® PT OVF Packet May be Lost if Immediately Preceding a TraceStop
SKZ45	X	X	No Fix	Debug Exceptions May Be Lost or Misreported Following WRMSR to IA32_BIOS_UPDT_TRIG Executed Immediately After MOV SS or POP SS
SKZ46	X	X	No Fix	x87 FDP Value May be Saved Incorrectly
SKZ47	X	X	No Fix	The Intel® PT CR3 Filter is not Re-evaluated on VM Entry
SKZ48	X	X	No Fix	BNDLDX and BNDSTX May not Signal #GP on Non-Canonical Bound Directory Access
SKZ49	X	X	No Fix	Performance Monitor Event For Outstanding Offcore Requests And Snoop Requests May be Incorrect Performance Monitor Event for Outstanding Offcore Requests May be Incorrect
SKZ50	X	X	No Fix	Branch Instructions May Initialize MPX Bound Registers Incorrectly
SKZ51	X	X	No Fix	Execution of VAESIMC or VAESKEYGENASSIST with an Illegal Value For VEX.vvvv May Produce a #NM Exception
SKZ52	X	X	No Fix	Writing a Non-Canonical Value to an LBR MSR Does not Signal a #GP When Intel® PT is Enabled
SKZ53	X	X	No Fix	VM Entry that Clears TraceEn May Generate a FUP
SKZ54	X	X	No Fix	Spurious Corrected Errors May be Reported
SKZ55	X	X	No Fix	Some Bits in MSR_MISC_PWR_MGMT May be Updated on Writing Illegal Values to this MSR
SKZ56	X	X	No Fix	CTR_FRZ May not Freeze Some Counters
SKZ57	X	X	No Fix	Writing Non-Zero Values to Read Only Fields in IA32_THERM_STATUS MSR May #GP
SKZ58	X	X	No Fix	Debug Exceptions May be Lost in the Case Of Machine Check Exception
SKZ59	X	X	No Fix	Processor May Hang on Complex Sequence of Conditions
SKZ60	X	X	No Fix	Branch Instruction Address May be Incorrectly Reported on TSX Abort When Using MPX
SKZ61	X	X	No Fix	Setting Performance Monitoring IA32_PERF_GLOBAL_STATUS_SET MSR Bit 63 May not #GP
SKZ62	X	X	No Fix	Problematic Port Bit with Locked Transactions And P2P May Cause System Hang
SKZ63	X	X	No Fix	Processor May Hang when Executing Code in an HLE Transaction Region
SKZ64	X	X	No Fix	Advanced Error Reporting Error Indication Pins May not Clear After Warm Reset
SKZ65	X	X	No Fix	IDI_MISC Performance Monitoring Events May be Inaccurate
SKZ66	X	X	No Fix	Execution of VAESENCLAST Instruction May Produce #NM Exception Instead of #UD Exception
SKZ67	X	X	No Fix	Reading Some C-State Residency MSRs May Result in Unpredictable System Behavior
SKZ68	X	X	No Fix	Intel® MBA Read After MSR Write May Return Incorrect Value
SKZ69	X	X	No Fix	DRAM_ENERGY_STATUS May Report Higher Than Expected DRAM Power Consumption for CPU0 After a Warm Reset
SKZ70	X	X	No Fix	DDR4 Memory Bandwidth May be Lower than Expected at 2133 and 1866 MHz
SKZ71	X	X	No Fix	VccSA Voltage May Increase After a Demoted Warm Reset Cycle
SKZ72	X	X	No Fix	Intel® MBA May Incorrectly Throttle all Threads
SKZ73	X	X	No Fix	VCVTPS2PH To Memory May Update MXCSR in the Case of a Fault on the Store
SKZ74	X	X	No Fix	Intel® PT May Drop All Packets After an Internal Buffer Overflow
SKZ75	X	X	No Fix	An IERR May be Seen when the CPU Attempts Consecutive C6 Entries
SKZ76	X	X	No Fix	Non-Zero Values May Appear in ZMM Upper Bits After SSE Instructions
SKZ77	X	X	No Fix	ZMM/YMM Registers May Contain Incorrect Values
SKZ78	X	X	No Fix	Intel® PT CYC Packet Can be Dropped When Immediately Preceding PSB

Table 1. Errata Summary Table (Sheet 4 of 4)

Number	Steppings		Status	Errata
	U-0	M-0		
SKZ79	X	X	No Fix	Intel® PT ToPA Tables Read From Non-Cacheable Memory During an Intel® TSX Transaction May Lead to Processor Hang
SKZ80	X	X	No Fix	Performing an XACQUIRE to an Intel® PT ToPA Table May Lead to Processor
SKZ81	X	X	No Fix	Intel® PT PSB+ Packets May be Omitted on a C6 Transition
SKZ82	X	X	No Fix	Intel® PT PacketEn Change on C-state Wake May Not Generate a TIP Packet
SKZ83	X	X	No Fix	When Virtualization Exceptions are Enabled, EPT Violations May Generate Erroneous Virtualization Exceptions
SKZ84	X	X	No Fix	Use of VMX TSC Scaling or TSC Offsetting Will Result in Corrupted Intel PT Packets
SKZ85	X	X	No Fix	Using Intel® TSX Instructions May Lead to Unpredictable System Behavior
SKZ86	X	X	No Fix	Performance in an 8sg System May Be Lower Than Expected
SKZ87	X	X	No Fix	Performance Monitoring General Purpose Counter 3 May Contain Unexpected Values
SKZ88	X	X	No Fix	Memory May Continue to Throttle after MEMHOT# De-assertion
SKZ89	X	X	No Fix	Unexpected Uncorrected Machine Check Errors May Be Reported
SKZ90	X	X	No Fix	Intel® PT Trace May Drop Second Byte of CYC Packet
SKZ91	X	X	No Fix	IMC Patrol Scrubbing Engine May Hang
SKZ92	X	X	No Fix	Executing Some Instructions May Cause Unpredictable Behavior
SKZ93	X	X	No Fix	Intel® MBM Counters May Report System Memory Bandwidth Incorrectly
SKZ94	X	X	No Fix	Processor May Behave Unpredictably on Complex Sequence of Conditions Which Involve Branches That Cross 64 Byte Boundaries
SKZ95	X	X	No Fix	Voltage/Frequency Curve Transitions May Result in Machine Check Errors or Unpredictable System Behavior
SKZ96	X	X	No Fix	DMI and PCIe Interfaces May See Elevated Bit Error Rates
SKZ97	X	X	No Fix	Unexpected Page Faults in Guest Virtualization Environment
SKZ98	X	X	No Fix	STIBP May Not Function as Intended
SKZ99	X	X	No Fix	Instruction Fetch May Cause Machine Check if Page Size Was Changed Without Invalidation

Specification Changes

Number	Specification Changes
	None for this revision of this specification update.

Specification Clarifications

No.	Specification Clarifications
	None for this revision of this specification update.

Documentation Changes

No.	Documentation Changes
	None for this revision of this specification update.

Errata

SKZ1 A CAP Error While Entering Package C6 May Cause DRAM to Fail to Enter Self-Refresh

Problem: A CAP (Command/Address Parity) error that occurs on the command to direct DRAM to enter self-refresh may cause the DRAM to fail to enter self-refresh although the processor enters-Package-C6.

Implication: Due to this erratum, DRAM may fail to be refreshed, which may result in uncorrected errors being reported from the DRAM.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ2 PCIe* Lane Error Status Register May Log False Correctable Error

Problem: Due to this erratum, PCIe* LNERRSTS (Device 0; Function 0; Offset 258h; bits [3:0]) may log false lane-based correctable errors.

Implication: Diagnostics cannot reliably use LNERRSTS to report correctable errors.

Workaround: None Identified

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ3 In Memory Mirror Mode, DataErrorChunk Field May be Incorrect

Problem: In Memory Mirror Mode, DataErrorChunk bits (IA32_MC7_MISC register MSR(41FH) bits [61:60]) may not correctly report the chunk containing an error.

Implication: Due to this erratum, this field is not accurate when Memory Mirror Mode is enabled.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ4 Intel® RDT MBM Does Not Accurately Track Write Bandwidth

Problem: Intel® RDT (Resource Director Technology) MBM (Memory Bandwidth Monitoring) does not count cacheable write-back traffic to local memory. This will result in the RDT MBM feature under counting total bandwidth consumed.

Implication: Applications using this feature may report incorrect memory bandwidth.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ5 PCIe* Port May Incorrectly Log Malformed_TLP Error

Problem: If the PCIe port receives a TLP that triggers both a Malformed_TLP error and an ECRC_TLP error, the processor should only log an ECRC_TLP error. However, the processor logs both errors.

Implication: Due to this erratum, the processor may incorrectly log Malformed_TLP errors.

Workaround: None Identified

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ6 Short Loops Which Use AH/BH/CH/DH Registers May Cause Unpredictable System Behavior

Problem: Under complex micro-architectural conditions, short loops of less than 64 instructions that use AH, BH, CH or DH registers as well as their corresponding wider register (e.g. RAX, EAX or AX for AH) may cause unpredictable system behavior. This can only happen when both logical processors on the same physical processor are active.

Implication: Due to this erratum, the system may experience unpredictable system behavior

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

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ7 Credits Not Returned For PCIe* Packets That Fail ECRC Check

Problem: The processor's IIO does not return credits back to the PCIe* link in case of end-to-end CRC (ECRC) errors.

Implication: Due to this erratum, the link may experience degraded performance or may eventually fail due to a loss of credits.

Workaround: For processors that support LER (Live Error Recovery) the link would be reset and credits would be restored. Processors that do not support LER should configure ECRC errors to be fatal.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ8 Link Training Error Due to Single Polarity of a PCIe* Differential Data Pair Being Disconnected

Problem: A PCIe Port may not reach L0 state if a single polarity of a PCIe* differential data pair is disconnected.

Implication: Due to this erratum, the Port will not downlink and be able to train up to L0.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ9 IODC Entry 0 Cannot be Masked

Problem: The Individual IODC (IO Directory Cache) Entry 0 cannot be masked using HA_COH_CFG_1, (Bus 1; Devices 11-8; Functions 7-0, Offset 0x11C, bit 0) therefore entry 0 is always allocated.

Implication: No functional implications.

Workaround: None.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ10 With eMCA2 Enabled a 3-Strike May Cause an Unnecessary CATERR# Instead of Only MSMI

Problem: When eMCA2 is enabled to cause an MSMI due to a 3-strike event, a pulsed CATERR# and MSMI# event may both be observed on the pins.

Implication: When this erratum occurs, an unnecessary CATERR# pulse may be observed.

Workaround: None.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ11 CMCI May not be Signaled for Corrected Error

Problem: Machine check banks 9, 10, and 11 may not signal CMCI after the first corrected error is reported in the bank even if the MCi_STATUS register has been cleared.

Implication: After the first corrected error is reported in one of the affected machine check banks, subsequent errors will be logged but may not result in a CMCI.

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

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ12 CSRs SVID and SDID are not Implemented For Some DDRIO And PCU Devices

Problem: The DDRIO (Bus: 3; Device 19,22; Function 6,7 and “Bus: 0; Device: 20,23; Function: 4,5,6,7;) and PCU (Bus: 3; Device 31; Functions 0,2) do not implement the SVID (Offset 0x2C) and SDID (Offset 0x2E) CSRs.

Implication: SW relying on DDRIO and PCU SVID and SDID CSR support may not function correctly.

Workaround: None. Do not use SVID and SDID for these devices and functions.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ13 Register Broadcast Read From DDRIO May Return a Zero Value

Problem: When performing a BIOS broadcast register read to DDRIO a value of 0 is always returned.

Implication: When this erratum occurs, BIOS may not be able to proceed due to always reading a value of 0.

Workaround: None. Use unicast register read for each instance instead of broadcast register read for all instances at once.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ14 Intel® CMT Counters May Not Count Accurately

Problem: Under complex microarchitectural conditions, the CMT (Cache Monitoring Technology) counters may overcount.

Implication: Software relying on CMT registers to enable resource allocation may not operate correctly. This may lead to reporting of more cachelines used than the cache supports or the counter wrapping and returning a too small value. WBINVD may not result in the CMT counters being zeroed. Intel has not observed this erratum in commercially available software.

Workaround: None.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ15 Intel® CAT May Not Restrict Cacheline Allocation Under Certain Conditions

Problem: Under certain microarchitectural conditions involving heavy memory traffic, cachelines may fill outside the allocated L3 capacity bitmask (CBM) associated with the current Class of Service (CLOS).

Implication: CAT (Cache Allocation Technology) may appear less effective at protecting certain classes of applications, including cache-sensitive workloads than on previous platforms.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ16 Intel® PCIe* Corrected Error Threshold Does Not Consider Overflow Count When Incrementing Error Counter

Problem: The PCIe* corrected error counter feature does not take the overflow bit in the count (bit 15 of XPCORERRCOUNTER (Bus; RootBus Device; 0 Function; 0 Offset; 4D0h)) into account when comparing the count to the threshold in XPCORERRTHRESHOLD.ERROR_THRESHOLD. Therefore, you end up with another interrupt once the counter has rolled over and hit your threshold + 0x8000.

Implication: Due to this erratum, the PCIe* corrected error signaling may occur even after the error count has exceeded the corrected error count threshold, not just a single time when reaching the threshold. Intel has not observed this erratum with any commercially available system.

Workaround: None Identified

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ17 IIO RAS VPP Hangs During The Warm Reset Test

Problem: When VPPCL bit 0 of VPP_reset_Mode (Bus 1; Device 30; Function 5; Offset 0xF0) bit is set to 0, and the CPU is undergoing reset flow while PCIe* hotplug operation is in process, the VPP (Virtual Pin Port) hotplug commands may stop responding.

Implication: Due to this erratum, during CPU reset hotplug commands may not get completed.

Workaround: None. Do not set VPP reset mode to 0.

SKZ18 For the Steppings affected, refer the *Summary Tables of Changes*.**Processor May Hang on Complex Sequence of Conditions**

Problem: A complex set of architectural and micro-architectural conditions may lead to a processor hang with an internal timeout error (MCACOD 0400H) logged into IA32_MCi_STATUS. When both logical processors in a core are active, this erratum will not occur unless there is no store on one of the logical processors for more than 10 seconds.

Implication: This erratum may result in a processor hang. Intel has not observed this erratum with any commercially available software.

Workaround: None Identified

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ19 Intel® PCIe* Root Port Electromechanical Interlock Control Register Can be Written

Problem: Electromechanical Interlock Control (bit 11) in the Slot Control register (B: Root Port; D: 0-3; F: 0 bits offset 0x18) in the PCIe* Capability table should be read-only and always return 0. Due to this erratum, this register can be written.

Implication: Writes to this bit can cause later reads to return the written value. However, this has no other effect on functionality.

Workaround: None Identified

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ20 System Hangs May Occur When IPQ And IRQ Requests Happen at the Same Time

Problem: When IPQ and IRQ requests happen at the same time, and the IPQ request is starved due to PAMatch/NotAllowSnoop on a TORID (Table of Request ID) then the IRQ request that is waiting for the TORID's SF/LLC may become invalid.

Implication: Due to this erratum, if IPQ and IRQ requests do not need to snoop any cores, then IPQ requests may block IRQ requests resulting in a system hang. Intel has only observed this erratum in a synthetic test environment.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes.*

SKZ21 Masked Bytes in a Vector Masked Store Instructions May Cause Write Back of a Cache Line

Problem: Vector masked store instructions to WB (write-back) memory-type that cross cache lines may lead to CPU writing back cached data even for cache lines where all of the bytes are masked.

Implication: The processor may generate writes of un-modified data. This can affect MMIO (Memory Mapped IO) or non-coherent agents in the following ways.
1. 1. For MMIO range that is mapped as WB memory type, this erratum may lead to MCE (Machine Check Exception) due to writing back data into the MMIO space. This applies only to cross page vector masked stores where one of the pages is in MMIO range
2. If the CPU cached data is stale, for example in the case of memory written directly by a non-coherent agent (agent that uses non-coherent writes), this erratum may lead to writing back stale cached data even if these bytes are masked.

Workaround: Platforms should not map MMIO memory space or non-coherent device memory space as WB memory. If WB is used for MMIO range, software or VMM should not map such MMIO page adjacent to a regular WB page (adjacent on the linear address space, before or after the IO page). Memory that may be written by non-coherent agents should be separated by at least 64 bytes from regular memory used for other purposes (on the linear address space).

Status: For the Steppings affected, refer the *Summary Tables of Changes.*

SKZ22 ERROR_N[2:0] Pins May not be Cleared After a Warm Reset

Problem: The processor’s ERROR_N[2:0] pins may not be cleared after a warm reset.

Implication: Due to this erratum, the ERROR_N[2:0] pins may incorrectly indicate a pending error after a warm reset.

Workaround: BIOS can contain code changes to work around this erratum.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ23 Intel® PCIe* Slot Presence Detect and Presence Detect Changed Logic not PCIe* Specification Compliant

Problem: When Hot-Plug Surprise is set in the Slot Capabilities register (Bus: RootBus, Dev: 1-3, Function: 0, Offset: A4h, Bit: 5), the Presence Detect State and Presence Detect Change in the Slot Status register (Bus: RootBus, Dev: 1-3, Function: 0, Offset: A2h), incorrectly ignores the out-of-band presence detect mechanism and only reflects the Physical Layer in-band presence detect mechanism.

Implication: Due to this erratum, if the Hot-Plug Surprise bit is set in the Slot Capabilities register, software will not be able to detect the presence of an adapter inserted while a slot is powered down. Therefore, Hot-Plug Surprise must only be set in configurations where the slot power is always enabled.

Workaround: None Identified

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ24 Debug Exceptions May be Lost or Misreported When MOV SS or POP SS Instruction is not Followed by a Write to SP

Problem: If a MOV SS or POP SS instruction generated a debug exception, and is not followed by an explicit write to the stack pointer (SP), the processor may fail to deliver the debug exception or, if it does, the DR6 register contents may not correctly reflect the causes of the debug exception.

Implication: Debugging software may fail to operate properly if a debug exception is lost or does not report complete information. Intel has not observed this erratum with any commercially available software.

Workaround: Software should explicitly write to the stack pointer immediately after executing MOV SS or POP SS.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ25 Incorrect Branch Predicted Bit in BTS/BTM Branch Records

Problem: BTS (Branch Trace Store) and BTM (Branch Trace Message) send branch records to the Debug Store management area and system bus respectively. The Branch Predicted bit (bit 4 of eighth byte in BTS/BTM records) should report whether the most recent branch was predicted correctly. Due to this erratum, the Branch Predicted bit may be incorrect.

Implication: BTS and BTM cannot be used to determine the accuracy of branch prediction.

Workaround: None Identified

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ26 DR6.B0-B3 May not Report all Breakpoints Matched When a MOV/POP SS is Followed by a Store or an MMX Instruction

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

Problem: store instruction.

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 (i.e., following them only with an instruction that writes

Implication: (E/R)SP).

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ27 Intel® PT TIP.PGD May Not Have Target IP Payload

Problem: When Intel PT (Intel® Processor Trace) is enabled and a direct unconditional branch clears IA32_RTIT_STATUS.FilterEn (MSR 571H, bit 0), due to this erratum, the resultingTIP.PGD (Target IP Packet, Packet Generation Disable) may not have an IP payload with the target IP.

Implication: It may not be possible to tell which instruction in the flow caused the TIP.PGD using only the information in trace packets when this erratum occurs.

Workaround: The Intel PT trace decoder can compare direct unconditional branch targets in the source with the FilterEn address range(s) to determine which branch cleared FilterEn.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ28 The Corrected Error Count Overflow Bit in IA32_ MC0_STATUS is not Updated When the UC Bit is Set

Problem: After a UC (uncorrected) error is logged in the IA32_MC0_STATUS MSR (401H), corrected errors will continue to be counted in the lower 14 bits (bits 51:38) of the Corrected Error Count. Due to this erratum, the sticky count overflow bit (bit 52) of the Corrected Error Count will not get updated when the UC bit (bit 61) is set to 1.

Implication: The Corrected Error Count Overflow indication will be lost if the overflow occurs after an uncorrectable error has been logged.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ29 SMRAM State-Save Area Above the 4GB Boundary May Cause Unpredictable System Behavior

Problem: If BIOS uses the RSM instruction to load the SMBASE register with a value that would cause any part of the SMRAM state-save area to have an address above 4-GBytes, subsequent transitions into and out of SMM (system-management mode) might save and restore processor state from incorrect addresses.

Implication: This erratum may cause unpredictable system behavior. Intel has not observed this erratum with any commercially available system.

Workaround: Ensure that the SMRAM state-save area is located entirely below the 4GB address boundary.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ30 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, refer the *Summary Tables of Changes*.

SKZ31 x87 FPU Exception (#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 executing when an Enhanced Intel SpeedStep® Technology transitions, an Intel® Turbo Boost Technology transitions, or a Thermal Monitor events occurs, the #MF may be taken 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, refer the Summary Tables of Changes.

SKZ32 POPCNT Instruction May Take Longer to Execute than Expected

Problem: POPCNT instruction execution with a 32 or 64 bit operand may be delayed until previous non-dependent instructions have executed.

Implication: Software using the POPCNT instruction may experience lower performance than expected.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes.*

SKZ33 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® HyperThreading 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, refer the *Summary Tables of Changes*.

SKZ34 Intel® Processor Trace PSB+ Packets May Contain Unexpected Packets

Problem: Some Intel Processor Trace packets should be issued only between TIP.PGE (Target IP Packet.Packet Generation Enable) and TIP.PGD (Target IP Packet.Packet Generation Disable) packets. Due to this erratum, when a TIP.PGE packet is generated it may be preceded by a PSB+ (Packet Stream Boundary) that incorrectly includes FUP (Flow Update Packet) and MODE.Exec packets.

Implication: Due to this erratum, FUP and MODE.Exec may be generated unexpectedly.

Workaround: Decoders should ignore FUP and MODE.Exec packets that are not between TIP.PGE and TIP.PGD packets.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ35 Performance Monitoring Counters May Undercount When Using CPL Filtering

Problem: Performance Monitoring counters configured to count only OS or only USR events (by setting only one of bits 16 or 17 in IA32_PERFEVTSELx) may undercount for a short cycle period of typically less than 100 processor clock cycles after the processor transitions to a new CPL. Events affected may include those counting CPL transitions (by additionally setting the edge-detect bit 18 in IA32_PERFEVTSELx).

Implication: Due to this erratum, Performance Monitoring counters may report counts lower than expected.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ36 Intel® PT ToPA PMI Does not Freeze Performance Monitoring Counters

Problem: Due to this erratum, if IA32_DEBUGCTL.FREEZE_PERFMON_ON_PMI (MSR 1D9H, bit12) is set to 1 when Intel PT (Processor Trace) triggers a ToPA (Table of Physical Addresses) PMI (PerfMon Interrupt), performance monitoring counters are not frozen as expected.

Implication: Performance monitoring counters will continue to count for events that occur during PMI handler execution.

Workaround: PMI handler software can programmatically stop performance monitoring counters upon entry.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ37 Performance Monitoring Load Latency Events May be Inaccurate For Gather Instructions

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 (an extension of PEBS). However due to this erratum, these events may count incorrectly for VGATHER*/VPGATHER* instructions.

Implication: The Load Latency Performance Monitoring events may be Inaccurate for Gather instructions.

Workaround: None Identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ38 CPUID TLB Associativity Information is Inaccurate

Problem: CPUID leaf 2 (EAX=02H) TLB information inaccurately reports that the shared 2nd-Level TLB is 6-way set associative (value C3H), although it is 12-way set associative. Other information reported by CPUID leaf 2 is accurate.

Implication: Software that uses CPUID shared 2nd-level TLB associativity information for value C3H may operate incorrectly. Intel has not observed this erratum to impact the operation of any commercially available software.

Workaround: None identified. Software should ignore the shared 2nd-Level TLB associativity information reported by CPUID for the affected processors.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ39 Vector Masked Store Instructions May Cause Write Back of Cache Line Where Bytes are Masked

Problem: Vector masked store instructions to WB (write-back) memory-type that cross cache lines may lead to CPU writing back cached data even for cache lines where all of the bytes are masked.
This can affect MMIO (Memory Mapped IO) or non-coherent agents in the following ways:

- For MMIO range that is mapped as WB memory type, this erratum may lead to MCE

(Machine Check Exception) due to writing back data into the MMIO space. This applies only to cross page vector masked stores where one of the pages is in MMIO range.
• If the CPU cached data is stale, for example in the case of memory written directly by a non-coherent agent (agent that uses non-coherent writes), this erratum may lead to writing back stale cached data even if these bytes are masked.

Implication: CPU may generate writes into MMIO space which lead to MCE, or may write stale data into memory also written by non-coherent agents.

Workaround: It is recommended not to map MMIO range as WB. If WB is used for MMIO range, OS or VMM should not map such MMIO page adjacent to a regular WB page (adjacent on the linear address space, before or after the IO page). Memory that may be written by non-coherent agents should be separated by at least 64 bytes from regular memory used for other purposes (on the linear address space).

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ40 Incorrect FROM_IP Value For an RTM Abort in BTM or BTS May be Observed

Problem: During RTM (Restricted Transactional Memory) operation when branch tracing is enabled using BTM (Branch Trace Message) or BTS (Branch Trace Store), 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, refer the *Summary Tables of Changes*.

SKZ41 PEBS Record After a WRMSR to IA32_BIOS_UPDT_TRIG May be Incorrect

Problem: A PEBS record generated by a WRMSR to IA32_BIOS_UPDT_TRIG MSR (79H) may have an incorrect value in the Eventing EIP field if an instruction prefix was used on the WRMSR.

Implication: The Eventing EIP field of the generated PEBS record may be incorrect. Intel has not observed this erratum with any commercially available software.

Workaround: Instruction prefixes have no architecturally-defined function for the WRMSR instruction; instruction prefixes should not be used with the WRMSR instruction.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ42 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 to a different cache line.

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

Workaround: Software should not rely on a locked instruction to fence subsequent executions of MOVNTDQA. Software should insert an MFENCE instruction if it needs to preserve order between streaming loads and other memory operations.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ43 #GP on Segment Selector Descriptor that Straddles Canonical Boundary May Not Provide Correct Exception Error Code

Problem: During a #GP (General Protection 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, refer the *Summary Tables of Changes*.

SKZ44 Intel® PT OVF Packet May be Lost if Immediately Preceding a TraceStop

Problem: If an Intel PT (Intel® Processor Trace) internal buffer overflow occurs immediately before software executes a taken branch or event that enters an Intel PT TraceStop region, the OVF (Overflow) packet may be lost.

Implication: The trace decoder will not view the OVF packet, nor any subsequent packets (e.g., TraceStop) that were lost due to overflow.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ45 Debug Exceptions May Be Lost or Misreported Following WRMSR to IA32_BIOS_UPDT_TRIG Executed Immediately After MOV SS or POP SS

Problem: If the WRMSR instruction writes to IA32_BIOS_UPDT_TRIG (79H) immediately after an execution of MOV SS or POP SS that generated a debug exception, the processor may fail to deliver the debug exception or, if it does, the DR6 register contents may not correctly reflect the causes of the debug exception.

Implication: Debugging software may fail to operate properly if a debug exception is lost or does not report complete information. Intel has not observed this erratum with any commercially available software.

Workaround: Software should avoid using WRMSR instruction immediately after executing MOV SS or POP SS.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ46 x87 FDP Value May be Saved Incorrectly

Problem: Execution of the FSAVE, FNSAVE, FSTENV, or FNSTENV instructions in real-address mode or virtual-8086 mode may save an incorrect value for the x87 FDP (FPU data pointer). This erratum does not apply if the last non-control x87 instruction had an unmasked exception.

Implication: Software operating in real-address mode or virtual-8086 mode that depends on the FDP value for non-control x87 instructions without unmasked exceptions may not operate properly. Intel has not observed this erratum in any commercially available software.

Workaround: None identified. Software should use the FDP value saved by the listed instructions only when the most recent non-control x87 instruction incurred an unmasked exception.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ47 The Intel® PT CR3 Filter is not Re-evaluated on VM Entry

Problem: On a VMRESUME or VMLAUNCH with both TraceEn[0] and CR3Filter[7] in IA32_RTIT_CTL (MSR 0570H) set to 1 both before the VM Entry and after, the new value of CR3 is not compared with IA32_RTIT_CR3_MATCH (MSR 0572H).

Implication: The Intel® PT (Processor Trace) CR3 filtering mechanism may continue to generate packets despite a mismatching CR3 value, or may fail to generate packets despite a matching CR3, as a result of an incorrect value of IA32_RTIT_STATUS.ContextEn[1] (MSR 0571H) that results from the failure to re-evaluate the CR3 match on VM entry.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ48 BNDLDX and BNDSTX May not Signal #GP on Non-Canonical Bound Directory Access

Problem: BNDLDX and BNDSTX instructions access the bound's directory and table to load or store bounds. These accesses should signal #GP (general protection exception) when the address is not canonical (i.e., bits 48 to 63 are not the sign extension of bit 47). Due to this erratum, #GP may not be generated by the processor when a non-canonical address is used by BNDLDX or BNDSTX for their bound directory memory access.

Implication: Intel has not observed this erratum with any commercially available software.

Workaround: Software should use canonical addresses for bound directory accesses.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ49 Performance Monitor Event For Outstanding Offcore Requests And Snoop Requests May be Incorrect Performance Monitor Event for Outstanding Offcore Requests May be Incorrect

Problem: The performance monitor event OFFCORE_REQUESTS_OUTSTANDING (Event 60H, any Umask Value) should count the number of offcore outstanding transactions each cycle. Due to this erratum, the counts may be higher or lower than expected.

Implication: The performance monitor event OFFCORE_REQUESTS_OUTSTANDING may reflect an incorrect count.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ50 Branch Instructions May Initialize MPX Bound Registers Incorrectly

Problem: Depending on the current Intel® MPX (Memory Protection Extensions) configuration, execution of certain branch instructions (near CALL, near RET, near JMP, and Jcc instructions) without a BND prefix (F2H) initialize the MPX bound registers. Due to this erratum, execution of such a branch instruction on a user-mode page may not use the MPX configuration register appropriate to the current privilege level (BNDCFGU for CPL 3 or BNDCFGS otherwise) for determining whether to initialize the bound registers; it may thus initialize the bound registers when it should not, or fail to initialize them when it should.

Implication: After a branch instruction on a user-mode page has executed, a #BR (bound-range) exception may occur when it should not have or a #BR may not occur when one should have.

Workaround: If supervisor software is not expected to execute instructions on user-mode pages, software can avoid this erratum by setting CR4.SMEP[bit 20] to enable supervisormode execution prevention (SMEP). If SMEP is not available or if supervisor software is expected to execute instructions on user-mode pages, no workaround is identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ51 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 a #UD (Invalid-Opcode) 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 #NM (Device-Not-Available) 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, refer the *Summary Tables of Changes*.

SKZ52 Writing a Non-Canonical Value to an LBR MSR Does not Signal a #GP When Intel® PT is Enabled

Problem: If Intel PT (Intel Processor Trace) is enabled, WRMSR will not cause a general protection exception (#GP) on an attempt to write a non-canonical value to any of the following MSRs:
— MSR_LASTBRANCH_{0 - 31}_FROM_IP (680H - 69FH)
— MSR_LASTBRANCH_{0 - 31}_TO_IP (6C0H - 6DFH)

— MSR_LASTBRANCH_FROM_IP (1DBH)
— MSR_LASTBRANCH_TO_IP (1DCH)
— MSR_LASTINT_FROM_IP (1DDH)
— MSR_LASTINT_TO_IP (1DEH)

Instead the same behavior will occur as if a canonical value had been written. Specifically, the WRMSR will be dropped and the MSR value will not be changed.

Implication: Due to this erratum, an expected #GP may not be signaled.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ53 VM Entry that Clears TraceEn May Generate a FUP

Problem: Problem: If VM entry clears Intel® PT (Intel Processor Trace) IA32_RTIT_CTL.TraceEn (MSR 570H, bit 0) while PacketEn is 1 then a FUP (Flow Update Packet) will precede the TIP.PGD (Target IP Packet, Packet Generation Disable). VM entry can clear TraceEn if the VM-entry MSR-load area includes an entry for the IA32_RTIT_CTL MSR.

Implication: When this erratum occurs, an unexpected FUP may be generated that creates the appearance of an asynchronous event taking place immediately before or during the VM entry.

Workaround: The Intel PT trace decoder may opt to ignore any FUP whose IP matches that of a VM entry instruction.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ54 Spurious Corrected Errors May be Reported

Problem: Due to this erratum, spurious corrected errors may be logged in the IA32_MC0_STATUS MSR (401H) register with the valid field (bit 63) set, the uncorrected error field bit (bit 61) not set, a Model Specific Error Code (bits [31:16]) of 0x0001, and an MCA Error Code (bits [15:0]) of 0x0005. If CMCI is enabled, these spurious corrected errors also signal interrupts.

Implication: When this erratum occurs, software may view an unusual rate of spurious corrected errors that may interfere with reporting non-spurious corrected errors.

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

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ55 Some Bits in MSR_MISC_PWR_MGMT May be Updated on Writing Illegal Values to this MSR

Problem: Attempts to write illegal values to MSR_MISC_PWR_MGMT (MSR 0x1AA) result in #GP (General Protection Fault) and should not change the MSR value. Due to this erratum, some bits in the MSR may be updated on writing an illegal value.

Implication: Certain fields may be updated with allowed values when writing illegal values to MSR_MISC_PWR_MGMT. Such writes will always result in #GP as expected.

Workaround: None identified. Software should not attempt to write illegal values to this MSR.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ56 CTR_FRZ May not Freeze Some Counters

Problem: IA32_PERF_GLOBAL_STATUS.CTR_FRZ (MSR 38EH, bit 59) is set when either (1) IA32_DEBUGCTL.FREEZE_PERFMON_ON_PMI (MSR 1D9H, bit 12) is set and a PMI is triggered, or (2) software sets bit 59 of IA32_PERF_GLOBAL_STATUS_SET (MSR 391H). When set, CTR_FRZ should stop all core performance monitoring counters from counting. However, due to this erratum, IA32_PMC4-7 (MSR C5-C8H) may not stop counting. IA32_PMC4-7 are only available when a processor core is not shared by two logical processors.

Implication: Implication: General performance monitoring counters 4-7 may not freeze when IA32_PERF_GLOBAL_STATUS.CTR_FRZ is set.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ57 Writing Non-Zero Values to Read Only Fields in IA32_THERM_STATUS MSR May #GP

Problem: IA32_THERM_STATUS MSR (19CH) includes read-only (RO) fields as well as writable fields. Writing a non-zero value to any of the read-only fields may cause a #GP.

Implication: Due to this erratum, software that reads the IA32_THERM_STATUS MSR, modifies some of the writable fields, and attempts to write the MSR back may #GP.

Workaround: Software should clear all read-only fields before writing to this MSR.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ58 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 PCC (processor context corrupted) 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, refer the Summary Tables of Changes.

SKZ59 Processor May Hang on Complex Sequence of Conditions

Problem: A complex set of architectural and micro-architectural conditions may lead to a processor hang with an internal timeout error (MCACOD 0400H) logged into IA32_MC3_STATUS (MSR 040DH, bits [15:0]). When both logical processors in a core are active, this erratum will not occur in one logical processor unless there is no interrupt for more than 10 seconds to the other logical processor

Implication: This erratum may result in a processor hang. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes.*

SKZ60 Branch Instruction Address May be Incorrectly Reported on TSX Abort When Using MPX

Problem: When using Intel® Memory Protection Extensions (MPX), an Intel® Transactional Synchronization Extensions (TSX) transaction abort will occur in case of legacy branch (that causes bounds registers INIT) when at least one MPX bounds register was in a NON-INIT state. On such an abort, the branch Instruction address should be reported in the FROM_IP field in the Last Branch Records (LBR), Branch Trace Store (BTS) and Branch Trace Message (BTM) as well as in the Flow Update Packets (FUP) source IP address for Processor Trace (PT). Due to this erratum, the FROM_IP field in LBR/BTS/BTM, as well as the Flow Update Packets (FUP) source IP address that correspond to the TSX abort, may point to the preceding instruction.

Implication: Software that relies on the accuracy of the FROM_IP field / FUP source IP address and uses TSX may operate incorrectly when MPX is used.

Workaround: None identified.

Status: For the Steppings affected, refer the *Summary Tables of Changes*.

SKZ61 Setting Performance Monitoring IA32_PERF_GLOBAL_STATUS_SET MSR Bit 63 May not #GP

Problem: Bit 63 of IA32_PERF_GLOBAL_STATUS_SET MSR (391H) is reserved. Due to this erratum, setting the bit will not result in General Protection Fault (#GP).

Implication: Software that attempts to set bit 63 of IA32_PERF_GLOBAL_STATUS_SET MSR does not generate #GP. There are no other system implications to this behavior.

Workaround: None identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ62 Problematic Port Bit with Locked Transactions And P2P May Cause System Hang

Problem: When the Problematic_Port_for_Lock_Flows bit (bit 38) of the MISCCTRLSTS register (Rootbus 0; Device 0; Function 0; Offset 188H) for a given device is set and that device generates peer-to-peer traffic, locked transactions that target the device may lead to a processor hang.

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

Workaround: None Identified. Devices that require Problematic_Port_for_Lock_Flows bit (bit 38) to be set must not initiate peer-to-peer traffic.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ63 Processor May Hang when Executing Code in an HLE Transaction Region

Problem: Under certain conditions, if the processor acquires an HLE (Hardware Lock Elision) lock via the XACQUIRE instruction in the Host Physical Address range between 40000000H and 403FFFFFH, it may hang with an internal timeout error (MCACOD 0400H) logged into IA32_MCi_STATUS.

Implication: Due to this erratum, the processor may hang after acquiring a lock via XACQUIRE.

Workaround: BIOS can reserve the host physical address ranges of 40000000H and 403FFFFFH (example, map it as UC/MMIO). Alternatively, the VMM (Virtual Machine Monitor) can reserve that address range so no guest can use it. In non-virtualized systems, the OS can reserve that memory space.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ64 Advanced Error Reporting Error Indication Pins May not Clear After Warm Reset

Problem: The Advanced Error Reporting (AER) error indication pins (ERROR_N[2:0]) may not be cleared after a warm reset.

Implication: Due to this erratum, hardware may observe (ERROR_N[2:0]) pins being incorrectly asserted following a warm reset.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ65 IDI_MISC Performance Monitoring Events May be Inaccurate

Problem: The IDI_MISC.WB_UPGRADE and IDI_MISC.WB_DOWNGRADE performance monitoring events (Event FEH; UMask 02H and 04H) count cache lines written back to the L3 cache. counts cache lines evicted from the L2 cache. Due to this erratum, the per logical processor count may be incorrect when both logical processors on the same physical core are active. The aggregate count of both logical processors is not affected by this erratum.

Implication: IDI_MISC performance monitoring events may be inaccurate.

Workaround: None identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ66 Execution of VAESENCLAST Instruction May Produce #NM Exception Instead of #UD Exception

Problem: Execution of VAESENCLAST with VEX.L= 1 should signal a #UD (Invalid Opcode) exception, however, due to this erratum, a #NM (Device Not Available) exception may be signaled instead.

Implication: As a result of this erratum, an operating system may restore AVX and other state unnecessarily.

Workaround: None identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ67 Reading Some C-State Residency MSRs May Result in Unpredictable System Behavior

Problem: Under complex microarchitectural conditions, an MSR read of MSR_CORE_C3_RESIDENCY MSR (3FCh), MSR_CORE_C6_RESIDENCY MSR (3FDh), or MSR_CORE_C7_RESIDENCY MSR (3FEh) may result in unpredictable system behavior.

Implication: Unexpected exceptions or other unpredictable system behavior may occur.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ68 Intel® MBA Read After MSR Write May Return Incorrect Value

Problem: The MBA (Memory Bandwidth Allocation) feature defines a series of MSRs (0xD50-0xD57) to specify MBA Delay Values per Class of Service (CLOS), in the IA32_L2_QoS_Ext_BW_Thrtl_n MSR range. Certain values when written then read back may return an incorrect value in the MSR. Specifically, values greater than or equal to 10 (decimal) and less than 39 (decimal) written to the MBA Delay Value (Bits [15:0]) may be read back as 10%

Implication: The values written to the registers will be applied; however, software should be aware that an incorrect value may be returned.

Workaround: None identified

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ69 DRAM_ENERGY_STATUS May Report Higher Than Expected DRAM Power Consumption for CPU0 After a Warm Reset

Problem: After a warm reset, DRAM_ENERGY_STATUS (Bus: 1; Device: 30; Function: 2; Offset: 7C; Bit: [31:0]) for CPU0 may incorrectly indicate DRAM power consumption

Implication: Software that relies on the DRAM power consumption value may not behave as expected

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ70 DDR4 Memory Bandwidth May be Lower than Expected at 2133 and 1866 MHz

Problem: A DDR4 transaction credit imbalance between memory controllers may result in a lower than expected available memory bandwidth

Implication: Due to this erratum, DDR4 Memory Bandwidth may be lower than expected at 2133 and 1866 MHz

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ71 VccSA Voltage May Increase After a Demoted Warm Reset Cycle

Problem: VccSA may be increased after a Demoted Warm Reset.

Implication: Due to this erratum, VccSA may be higher than expected prior to a cold reset.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ72 Intel® MBA May Incorrectly Throttle all Threads

Problem: When one logical processor is disabled, the MBA (Memory Bandwidth Allocation) feature may select an incorrect MBA throttling value to apply to the core. A disabled logical processor may behave as though the Class of Service (CLOS) field in its associated IA32_PQR_ASSOC MSR (0xC8F) is set to zero (appearing to be set to CLOS[0]). When this occurs, the MBA throttling value associated with CLOS[0] may be incorrectly applied to both threads on the core

Implication: When Intel® Hyper-Threading technology is disabled or one logical thread on the core is disabled, the disabled thread is interpreted to have CLOS=0 set in its IA32_PQR_ASSOC MSR by hardware, which affects the calculation for the actual throttling value applied to the core. When this erratum occurs, the MBA throttling value associated with a given core may be incorrect

Workaround: To work around this erratum, CLOS[0] should not be used if any logical cores are disabled. Alternately, software may leave all threads enabled.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ73 VCVTPS2PH To Memory May Update MXCSR in the Case of a Fault on the Store

Problem: Execution of the VCVTPS2PH instruction with a memory destination may update the MXCSR exceptions flags (bits [5:0]) if the store to memory causes a fault (example, #PF) or VM exit. The value written to the MXCSR exceptions flags is what would have been written if there were no fault.

Implication: Software may view exceptions flags set in MXCSR, although the instruction has not successfully completed due to a fault on the memory operation. Intel has not observed this erratum to affect any commercially available software.

Workaround: None identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ74 Intel® PT May Drop All Packets After an Internal Buffer Overflow

Problem: Due to a rare microarchitectural condition, an Intel® PT (Processor Trace) ToPA (Table of Physical Addresses) entry transition can cause an internal buffer overflow that may result in all trace packets, including the OVF (Overflow) packet, being dropped.

Implication: When this erratum occurs, all trace data will be lost until either PT is disabled and re-enabled via IA32_RTIT_CTL.TraceEn [bit 0] (MSR 0570H) or the processor enters and exits a C6 or deeper C state.

Workaround: None identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ75 An IERR May be Seen when the CPU Attempts Consecutive C6 Entries

Problem: When the CPU attempts consecutive C6 entries, it may result in an Internal Timer Error Machine Check Event (MCACOD = 0x400).

Implication: Due to this erratum, the system may hang with an Internal Timer Error.

Workaround: A BIOS code change may be used as a workaround for this erratum. Adjust the delayed deep C-State timer via BIOS-to-PCODE mailbox setting. (Command = 0xbb; Data = 0x0c181836). This command should be issued by one thread per socket prior to CPL3 in the BIOS flow. An example of using this mailbox: 1. Set MSR 608h = 0x0c181836 2. Set MSR 607h = 0x800000bb 3. Then poll MSR 607h until bit 31 [RUN_BUSY] has cleared.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ76 Non-Zero Values May Appear in ZMM Upper Bits After SSE Instructions

Problem: Under complex microarchitectural conditions, a VGATHER instruction with ZMM16-31 destination register followed by an SSE instruction in the next 4 instructions, may cause the ZMM register that is aliased to the SSE destination register to have non-zero values in bits 256-511. This may happen only when ZMM0-15 bits 256-511 are all zero, and there are no other instructions that write to ZMM0-15 in between the VGATHER and the SSE instruction. Subsequent SSE instructions that write to the same register will reset the affected upper ZMM bits and XSAVE will not expose these ZMM values as long as no other AVX512 instruction writes to ZMM0-15. This erratum will not occur in software that uses VZEROUPPER between AVX instructions and SSE instructions as recommended in the SDM.

Implication: Due to this erratum, an unexpected value may appear in a ZMM register aliased to an SSE destination. Software may observe this value only if the ZMM register aliased to the SSE instruction destination is used and VZEROUPPER is not used between AVX and SSE instructions. Intel has not observed this erratum with any commercially available software

Workaround: None Identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ77 ZMM/YMM Registers May Contain Incorrect Values

Problem: Under complex microarchitectural conditions values stored in ZMM and YMM registers may be incorrect.

Implication: Due to this erratum, YMM and ZMM registers may contain an incorrect value. Intel® has not observed this erratum with any commercially available software.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ78 Intel® PT CYC Packet Can be Dropped When Immediately Preceding PSB

Problem: Due to a rare microarchitectural condition, generation of an Intel® PT (Processor Trace) PSB (Packet Stream Boundary) packet can cause a single CYC (Cycle Count) packet, possibly along with an associated MTC (Mini Time Counter) packet, to be dropped.

Implication: An Intel® PT decoder that is using CYCs to track time or frequency will get an improper value due to the lost CYC packet.

Workaround: If an Intel® PT decoder is using CYCs and MTCs to track frequency, and either the first MTC following a PSB shows that an MTC was dropped, or the CYC value appears to be 4095 cycles short of what is expected, the CYC value associated with that MTC should not be used. The decoder should wait for the next MTC before measuring frequency again.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ79 Intel® PT ToPA Tables Read From Non-Cacheable Memory During an Intel® TSX Transaction May Lead to Processor Hang

Problem: If an Intel® PT (Processor Trace) ToPA (Table of Physical Addresses) table is placed in UC (Uncacheable) or USWC (Uncacheable Speculative Write Combining) memory, and a ToPA output region is filled during an Intel® TSX (Transaction Synchronization) transaction, the resulting ToPA table read may cause a processor hang.

Implication: Placing Intel® PT ToPA tables in non-cacheable memory when Intel® TSX is in use may lead to a processor hang.

Workaround: None identified. Intel® PT ToPA tables should be located in WB memory if Intel® TSX is in use.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ80 Performing an XACQUIRE to an Intel® PT ToPA Table May Lead to Processor

Problem: If an XACQUIRE lock is performed to the address of an Intel® PT (Processor Trace) ToPA (Table of Physical Addresses) table, and that table is later read by the CPU during the HLE (Hardware Lock Elision) transaction, the processor may hang.

Implication: Accessing ToPA tables with XACQUIRE may result in a processor hang.

Workaround: None identified. Software should not access ToPA tables using XACQUIRE. An OS or hypervisor may wish to ensure all application or guest writes to ToPA tables to take page faults or EPT violations.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ81 Intel® PT PSB+ Packets May be Omitted on a C6 Transition

Problem: An Intel® PT (Processor Trace) PSB+ (Packet Stream Boundary+) set of packets may not be generated as expected when IA32_RTIT_STATUS.PacketByteCnt[48:32] (MSR 0x571) reaches the PSB threshold and a logical processor C6 entry occurs within the following one KByte of trace output.

Implication: After a logical processor enters C6, Intel® PT output may be missing PSB+ sets of packets.

Workaround: None identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ82 Intel® PT PacketEn Change on C-state Wake May Not Generate a TIP Packet

Problem: A TIP.PGE (Target IP, Packet Generation Enabled) or TIP.PGD (Target IP, Packet Generation Disabled) packet may not be generated if Intel ® PT (Processor Trace) PacketEn changes after IA32_RTIT_STATUS.FilterEn (MSR 571H, bit 0) is re-evaluated on wakeup from C6 or deeper sleep state.

Implication: When code enters or exits an IP filter region without a taken branch, tracing may begin or cease without proper indication in the trace output. This may affect trace decoder behavior.

Workaround: None identified. A trace decoder will need to skip ahead to the next TIP or FUP packet to determine the current IP.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ83 When Virtualization Exceptions are Enabled, EPT Violations May Generate Erroneous Virtualization Exceptions

Problem: An access to a GPA (guest-physical address) may cause an EPT-violation VM exit. When the "EPT-violation #VE" VM-execution control is 1, an EPT violation may cause a #VE (virtualization exception) instead of a VM exit. Due to this erratum, an EPT violation may erroneously cause a #VE when the "suppress #VE" bit is set in the EPT paging-structure entry used to map the GPA being accessed. This erratum does not apply when the "EPT-violation #VE" VM-execution control is 0 or when delivering an event through the IDT. This erratum applies only when the GPA in CR3 is used to access the root of the guest paging-structure hierarchy (or, with PAE paging, when the GPA in a PDPTE is used to access a page directory).

Implication: When using PAE paging mode, an EPT violation that should cause an VMexit in the VMM may instead cause a VE# in the guest. In other paging modes, in addition to delivery of the erroneous #VE, the #VE may itself cause an EPT violation, but this EPT violation will be correctly delivered to the VMM.

Workaround: A VMM may support an interface that guest software can invoke with the VMCALL instruction when it detects an erroneous #VE.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ84 Use of VMX TSC Scaling or TSC Offsetting Will Result in Corrupted Intel PT Packets

Problem: When Intel(R) PT (Processor Trace) is enabled within a VMX (Virtual Machine Extensions) guest, and TSC (Time Stamp Counter) offsetting or TSC scaling is enabled for that guest, by setting primary processor-based execution control bit 3 or secondary processor-based execution control bit 25, respectively, in the VMCS (Virtual Machine Control Structure) for that guest, any TMA (TSC/MTC Alignment) packet generated will have corrupted values in the CTC (Core Timer Copy) and FastCounter fields. Additionally, the corrupted TMA packet will be followed by a bogus data byte.

Implication: An Intel PT decoder will be confused when using the TMA packet to align cycle time with wall-clock time. The byte that follows the TMA will likely cause a decoder error for an unexpected or unrecognized packet.

Workaround: None identified. If a TMA packet with any reserved payload bits set is encountered by an Intel PT decoder it should be ignored, along with the byte that immediately follows it. Alternatively, Intel PT users may opt to disable MTC and TMA packets by clearing IA32_RTIT_CTL.MTCEn[bit 9].

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ85 Using Intel® TSX Instructions May Lead to Unpredictable System Behavior

Problem: Under complex micro-architectural conditions, software using Intel® TSX (Transactional Synchronization Extensions) 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. See Intel® White Paper "Performance Monitoring Impact of TSX Memory Ordering Issue" Doc ID#604224 or contact your Intel® Representative for more information.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ86 Performance in an 8sg System May Be Lower Than Expected

Problem: In 8sg (8-socket glueless) systems, certain workloads may generate a significant stream of accesses to remote nodes, leading to unexpected congestion in the processor's snoop responses.

Implication: Due to this erratum, 8sg system performance may be lower than expected.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ87 Performance Monitoring General Purpose Counter 3 May Contain Unexpected Values

Problem: When Restricted Transactional Memory (RTM) is supported (CPUID.07H.EBX.RTM [bit 11] = 1) and when TSX_FORCE_ABORT=0, Performance Monitor Unit (PMU) general purpose counter 3 (IA32_PMC3, MSR C4H and IA32_A_PMC3, MSR 4C4H) may contain unexpected values. Further, IA32_PREFEVTSEL3 (MSR 189H) may also contain unexpected configuration values.

Implication: Due to this erratum, software that uses PMU general purposes counter 3 may read an unexpected count and configuration.

Workaround: Software can avoid this erratum by writing 1 to bit 0 of TSX_FORCE_ABORT (MSR 10FH) which will cause all Restricted Transactional Memory (RTM) transactions to abort with EAX code 0. TSX_FORCE_ABORT MSR is available when CPUID.07H.EDX[bit 13]=1.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ88 Memory May Continue to Throttle after MEMHOT# De-assertion

Problem: When MEMHOT# is asserted by an external agent, the CPU may continue to throttle memory after MEMHOT# de-assertion.

Implication: When this erratum occurs, memory throttling occurs even after de-assertion of MEMHOT#.

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

Status: No Fix

SKZ89 Unexpected Uncorrected Machine Check Errors May Be Reported

Problem: In rare micro-architectural conditions, the processor may report unexpected machine check errors. When this erratum occurs, IA32_MC0_STATUS (MSR 401H) will have the valid bit set (bit 63), the uncorrected error bit set (bit 61), a model specific error code of 03H (bits [31:16]) and an MCA error code of 05H (bits [15:0]).

Implication: Due to this erratum, software may observe unexpected machine check exceptions.

Workaround: None Identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ90 Intel® PT Trace May Drop Second Byte of CYC Packet

Problem: Due to a rare micro-architectural condition, the second byte of a 2-byte CYC (Cycle Count) packet may be dropped without an OVF (Overflow) packet.

Implication: A trace decoder may signal a decode error due to the lost trace byte.

Workaround: None identified. A mitigation is available for this erratum. If a decoder encounters a multi-byte CYC packet where the second byte has bit 0 (Ext) set to 1, it should assume that 4095 cycles have passed since the prior CYC packet, and it should ignore the first byte of the CYC and treat the second byte as the start of a new packet.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ91 IMC Patrol Scrubbing Engine May Hang

Problem: Under rare microarchitectural conditions, the processor's Integrated Memory Controller (IMC) Patrol Scrubbing Engine may hang.

Implication: When this erratum occurs, IMC Patrol Scrubbing will cease. Intel has only observed this erratum in a synthetic test environment when testing with high rates of ECC errors.

Workaround: None identified.

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ92 Executing Some Instructions May Cause Unpredictable Behavior

Problem: Under complex micro-architectural conditions, executing an X87, AVX, or integer divide instruction may result in unpredictable system behavior.

Implication: When this erratum occurs, the system may behave unpredictably. Intel has not observed this erratum with any commercially available software.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ93 Intel® MBM Counters May Report System Memory Bandwidth Incorrectly

Problem: Intel® 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. Refer 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, refer the Summary Tables of Changes.

SKZ94 Processor May Behave Unpredictably on Complex Sequence of Conditions Which Involve Branches That Cross 64 Byte Boundaries

Problem: Under complex micro-architectural conditions involving branch instructions bytes that span multiple 64 byte boundaries (cross cache line), unpredictable system behavior may occur.

Implication: When this erratum occurs, the system may behave unpredictably.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ95 Voltage/Frequency Curve Transitions May Result in Machine Check Errors or Unpredictable System Behavior

Problem: Under complex microarchitecture conditions, during voltage/frequency curve transitions, 3-strike machine check errors or other unpredictable system behavior may occur due to an issue in the FIVR logic.

Implication: When this erratum occurs, the system may cause a 3 strike machine check error or other unpredictable system behavior.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ96 DMI and PCIe Interfaces May See Elevated Bit Error Rates

Problem: The Direct Media Interface (DMI) or Peripheral Component Interconnect Express (PCIe) interfaces may be subject to a high bit error rate.

Implication: Due to this erratum, an elevated rate of packet CRC errors may be observed on these interfaces which may lead to a Machine Check Error and/or may hang the system.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ97 Unexpected Page Faults in Guest Virtualization Environment

Problem: Under complex micro-architectural conditions, a virtualized guest could observe unpredictable system behavior.

Implication: When this erratum occurs, systems operating in a virtualization environment may exhibit unexpected page faults (double faults) leading to guest OS shutdown.

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ98 STIBP May Not Function as Intended

Problem: When the Single Thread Indirect Branch Predictors bit (IA32_SPEC_CTL[STIBP] (MSR 48H, bit 1)) is set on one logical processor, then under specific micro-architectural conditions, one logical processor may be able to control the predicted targets of indirect branches on the other logical processors.

Implication: Software relying on STIBP to mitigate against cross-thread speculative branch target injection may allow an attacker running on one logical processor to induce another logical processor on the same core to speculatively execute a disclosure gadget that could allow protected data to be inferred through a side-channel method called Branch Target Injection. This erratum does not affect processors with Hyper-Threading disabled or enabling the cross-thread protections of Indirect Branch Restricted Speculation bit (IA32_SPEC_CTL[IBRS] (MSR 48H, bit 0)).

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

Status: For the Steppings affected, refer the Summary Tables of Changes.

SKZ99 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, refer the Summary Tables of Changes.

§ §