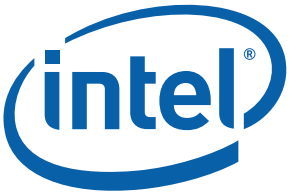


Intel® Core™ i7 Processor Family for the LGA-2011 Socket

Specification Update

Supporting Desktop Intel® Core™ i7-3960X and i7-3970X Extreme Edition Processor for the LGA-2011 Socket

Supporting Desktop Intel® Core™ i7-39xxK and i7-38xx Processor Series for the LGA-2011 Socket

May 2013

Notice: The products described in this document 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.

Document Number: 326198-012

Contents

§ §

Revision History

Revision	Description	Date
001	• Initial Release	November 2011
002	• Added Errata - BS92., BS93., BS94.	January 2012
003	• No update	February 2012
004	• Added Errata BS95 to BS169 • Removed Errata and replaced with new errata: BS37, BS46, BS49, BS52, BS53, BS64, BS73, BS76-BS86, BS89 • Updated Erratum: BS16	June 2012
005	• Added Errata BS170 to BS174	August 2012
006	• Added Erratum BS175	September 2012
007	• Added new processor i7-3970X	October 2012
008	• Added Erratum BS176	November 2012
009	• Added Errata BS177-BS180	December 2012
010	• Added documentation change BS1	January 2013
011	• Added Erratum BS181	April 2013
012	• Added new errata BS182-BS185	May 2013

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 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 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 / Location
Intel® Core™ i7 Processor Family for the LGA-2011 Socket Datasheet - Volume 1	326196-002
Intel® Core™ i7 Processor Family for the LGA-2011 Socket Datasheet - Volume 2	326197-002

Related Documents

Document Title	Document Number / Location
AP-485, Intel® Processor Identification and the CPUID Instruction	http://www.intel.com/design/processor/applnots/241618.htm
Intel® 64 and IA-32 Architecture Software Developer's Manual • *Volume 1: Basic Architecture* • *Volume 2A: Instruction Set Reference Manual A-M* • *Volume 2B: Instruction Set Reference Manual N-Z* • *Volume 3A: System Programming Guide* • *Volume 3B: System Programming Guide* • *IA-32 Intel® Architecture Optimization Reference Manual*	http://www.intel.com/products/processor/manuals/index.htm
Intel® 64 and IA-32 Architectures Software Developer's Manual Documentation Changes	http://download.intel.com/products/processor/manual/252046.pdf
Intel® 64 and IA-32 Architectures Optimization Reference Manual	http://www.intel.com/Assets/en_US/PDF/manual/248966.pdf

Notes:

1. Documentation changes for the Intel® 64 and IQ-32 Architecture Software Developer's Manual Volumes 1, 2A, 2B, 3A, and 3B and bug fixes are posted in the Intel® 64 and IA-32 Architecture Software Developer's Manual Documentation Changes. Follow the following link to become familiar with this file: http://www.intel.com/products/processor/manuals/index.htm

Nomenclature

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

Sightings are design defects or errors. These may cause the Product Name's 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

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 specification update when the appropriate changes are made to the appropriate product specification or user documentation (datasheets, manuals, etc.).

Identification Information

Component Identification using Programming Interface

The Intel® Core™ i7 processor family for the LGA-2011 Socket 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
31:28	27:20	19:16	15:14	13:12	11:8	7:4	3:0
	00000000b	0010b		00b	0110b	1101b	xxxxb
				B0	36S	8086	3405h
				B2	36S	8086h	3405h

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. See Table 1 for the processor stepping ID number in the CPUID information.

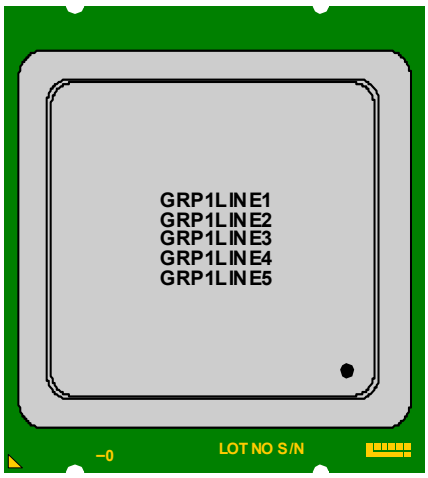
When EAX is initialized to a value of '1', the CPUID instruction returns the *Extended Family, Extended Model, Processor Type, Family Code, Model Number and Stepping ID* value in the EAX register. The EDX processor signature value after reset is equivalent to the processor signature output value in the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

Component Marking Information

The Intel® Core™ i7 processor family for the LGA-2011 Socket can be identified by the following component markings.

Figure 1. Intel® Core™ i7 Processor Family for the LGA-2011 Socket 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}

Table 1. Intel® Core™ i7 Processor Family for the LGA-2011 Socket Identification

S-Spec Number	Stepping	CPUID	Core Frequency (GHz)/ DDR3(MHz)	TDP (W)	# Cores	Cache Size (MB)	Notes
SR0GW	C-1	0X206D6	3.3/1600	130	6	15	
SR0H9	C-1	0X206D6	3.2/1600	130	6	12	
SR0KF	C-2	0X206D7	3.3/1600	130	6	15	
SR0KY	C-2	0X206D7	3.2/1600	130	6	12	
SR0WR	C-2	0X206D7	3.5/1600	150	6	15	

Summary Table of Changes

The table included in this section indicate the errata, Specification Changes, Specification Clarifications, or Document Changes which apply to the processor. 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.

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

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 a table row indicates this erratum is either new or modified from the previous version of the document.

Errata Number	Steppings		Status	ERRATA
	C-1	C-2		
BS1	X	X	No Fix	An Enabled Debug Breakpoint or Single Step Trap May Be Taken after MOV SS/ POP SS Instruction if it is Followed by an Instruction That Signals a Floating Point Exception
BS2	X	X	No Fix	APIC Error “Received Illegal Vector” May be Lost
BS3	X	X	No Fix	An Uncorrectable Error Logged in IA32_CR_MC2_STATUS May also Result in a System Hang
BS4	X	X	No Fix	B0-B3 Bits in DR6 For Non-Enabled Breakpoints May be Incorrectly Set
BS5	X	X	No Fix	Changing the Memory Type for an In-Use Page Translation May Lead to Memory-Ordering Violations
BS6	X	X	No Fix	Code Segment Limit/Canonical Faults on RSM May be Serviced before Higher Priority Interrupts/Exceptions and May Push the Wrong Address Onto the Stack
BS7	X	X	No Fix	Corruption of CS Segment Register During RSM While Transitioning From Real Mode to Protected Mode
BS8	X	X	No Fix	Debug Exception Flags DR6.B0-B3 Flags May be Incorrect for Disabled Breakpoints
BS9	X	X	No Fix	DR6 May Contain Incorrect Information When the First Instruction After a MOV SS,r/m or POP SS is a Store
BS10	X	X	No Fix	EFLAGS Discrepancy on Page Faults and on EPT-Induced VM Exits after a Translation Change
BS11	X	X	No Fix	Fault on ENTER Instruction May Result in Unexpected Values on Stack Frame
BS12	X	X	No Fix	Faulting MMX Instruction May Incorrectly Update x87 FPU Tag Word
BS13	X	X	No Fix	FREEZE_WHILE_SMM Does Not Prevent Event From Pending PEBS During SMM
BS14	X	X	No Fix	General Protection Fault (#GP) for Instructions Greater than 15 Bytes May be Preempted
BS15	X	X	No Fix	#GP on Segment Selector Descriptor that Straddles Canonical Boundary May Not Provide Correct Exception Error Code
BS16	X	X	No Fix	An Event May Intervene Before a System Management Interrupt That Results from IN or INS
BS17	X	X	No Fix	IRET under Certain Conditions May Cause an Unexpected Alignment Check Exception
BS18	X	X	No Fix	LER MSRs May Be Unreliable
BS19	X	X	No Fix	LBR, BTS, BTM May Report a Wrong Address when an Exception/Interrupt Occurs in 64-bit Mode
BS20	X	X	No Fix	MCi_Status Overflow Bit May Be Incorrectly Set on a Single Instance of a DTLB Error
BS21	X	X	No Fix	MONITOR or CLFLUSH on the Local XAPIC's Address Space Results in Hang
BS22	X	X	No Fix	MOV To/From Debug Registers Causes Debug Exception
BS23	X	X	No Fix	PEBS Record not Updated when in Probe Mode
BS24	X	X	No Fix	Performance Monitor Counter INST_RETIRED.STORES May Count Higher than Expected
BS25	X	X	No Fix	Performance Monitoring Event FP_MMX_TRANS_TO_MMX May Not Count Some Transitions

Errata Number	Steppings		Status	ERRATA
	C-1	C-2		
BS26	X	X	No Fix	REP MOVS/STOS Executing with Fast Strings Enabled and Crossing Page Boundaries with Inconsistent Memory Types may use an Incorrect Data Size or Lead to Memory-Ordering Violations.
BS27	X	X	No Fix	Reported Memory Type May Not Be Used to Access the VMCS and Referenced Data Structures
BS28	X	X	No Fix	Single Step Interrupts with Floating Point Exception Pending May Be Mishandled
BS29	X	X	No Fix	The Processor May Report a #TS Instead of a #GP Fault
BS30	X	X	No Fix	VM Exits Due to “NMI-Window Exiting” May Be Delayed by One Instruction
BS31	X	X	No Fix	Values for LBR/BTS/BTM Will be Incorrect after an Exit from SMM
BS32	X	X	No Fix	VPHMINPOSUW Instruction in VEX Format Does Not Signal #UD (Invalid Opcode Exception) When vex.vvvv !=1111
BS33	X	X	No Fix	Pending x87 FPU Exceptions (#MF) May be Signaled Earlier Than Expected
BS34	X	X	No Fix	VMREAD/VMWRITE Instruction May Not Fail When Accessing an Unsupported Field in VMCS
BS35	X	X	No Fix	Unexpected #UD on VZEROALL/VZEROUPPER
BS36	X	X	No Fix	Execution of Opcode 9BH with the VEX Opcode Extension May Produce a #NM Exception
BS37	X	X	No Fix	Enabling Opportunistic Self-Refresh and Pkg C2 State Can Severely Degrade PCIe* Bandwidth
BS38	X	X	No Fix	An Unexpected Page Fault or EPT Violation May Occur After Another Logical Processor Creates a Valid Translation for a Page
BS39	X	X	No Fix	Faulting Executions of XRSTOR May Update State Inconsistently
BS40	X	X	No Fix	Execution of FXSAVE or FXRSTOR With the VEX Prefix May Produce a #NM Exception
BS41	X	X	No Fix	Unexpected #UD on VPEXTRD/VPINSRD
BS42	X	X	No Fix	#GP May be Signaled When Invalid VEX Prefix Precedes Conditional Branch Instructions
BS43	X	X	No Fix	LBR, BTM or BTS Records May have Incorrect Branch From Information After an EIST/T-state/S-state/C1E Transition or Adaptive Thermal Throttling
BS44	X	X	No Fix	A Write to the IA32_FIXED_CTR1 MSR May Result in Incorrect Value in Certain Conditions
BS45	X	X	No Fix	L1 Data Cache Errors May be Logged With Level Set to 1 Instead of 0
BS46	X	X	No Fix	PECI RdPkgConfig() May Return Invalid Data For an Unsupported Channel
BS47	X	X	No Fix	For the affected steppings, see the Summary Tables of Changes.
BS48	X	X	No Fix	Interrupt From Local APIC Timer May Not Be Detectable While Being Delivered
BS49	X	X	No Fix	End Agent PCIe* Packet Errors May Result in a System Hang
BS50	X	X	No Fix	Poison Packets Will be Reported to PCIe* Port 1a When Forwarded to Port 1b
BS51	X	X	No Fix	IA32_MCi_ADDR Overwritten in The Case of Multiple Recoverable Instruction Fetch Errors
BS52	X	X	No Fix	PCIe* Link May Not Train to Full Width
BS53	X	X	No Fix	Spurious SMIs May Occur Due to MEMHOT# Assertion
BS54	X	X	No Fix	The PCIe* Current Compensation Value Default is Incorrect

Errata Number	Steppings		Status	ERRATA
	C-1	C-2		
BS55	X	X	No Fix	The PCIe* Link at 8.0 GT/s is Transitioning Too Soon to Normal Operation While Training
BS56	X	X	No Fix	A First Level Data Cache Parity Error May Result in Unexpected Behavior
BS57	X	X	No Fix	PECI Write Requests That Require a Retry Will Always Time Out
BS58	X	X	No Fix	The Vswing of the PCIe* Transmitter Exceeds The Specification
BS59	X	X	No Fix	When a Link is Degraded on a Port due to PCIe* Signaling Issues Correctable Receiver Errors May be Reported on The Neighboring Port
BS60	X	X	No Fix	A CMCI is Only Generated When the Memory Controller's Correctable Error Count Threshold is Exceeded
BS61	X	X	No Fix	PCIe* Rx DC Common Mode Impedance is Not Meeting the Specification
BS62	X	X	No Fix	A Modification to the Multiple Message Enable Field Does Not Affect The AER Interrupt Message Number field
BS63	X	X	No Fix	Unexpected PCIe* Set_Slot_Power_Limit Message on Writes to LNKCON
BS64	X	X	No Fix	PCIe* Link Bandwidth Notification Capability is Incorrect
BS65	X	X	No Fix	Locked Accesses Spanning Cachelines That Include PCI Space May Lead to a System Hang
BS66	X	X	No Fix	Cold Boot May Fail Due to Internal Timer Error
BS67	X	X	No Fix	PCIe* Rx Common Mode Return Loss is Not Meeting The Specification
BS68	X	X	No Fix	The Most Significant Bit of the CEC Cannot be Cleared Once Set
BS69	X	X	No Fix	PCIe* Adaptive Equalization May Not Train to the Optimal Settings
BS70	X	X	No Fix	A Core May Not Complete Transactions to The Caching Agent When C-States Are Enabled Leading to an Internal Timer Error
BS71	X	X	No Fix	TSC is Not Affected by Warm Reset
BS72	X	X	No Fix	Warm Resets May be Converted to Power-on Resets When Recovering From an IERR
BS73	X	X	No Fix	Port 3a Capability_Pointer Field is Incorrect When Configured in PCIe* Mode
BS74	X	X	No Fix	Processor May not Restore the VR12 DDR3 Voltage Regulator Phases upon Pkg C3 State Exit
BS75	X	X	No Fix	The Equalization Phase Successful Bits Are Not Compliant to The PCIe* Specification
BS76	X	X	No Fix	Four Outstanding PCIe* Configuration Retries May Cause Deadlock
BS77	X	X	No Fix	A PECI RdPciConfigLocal Command Referencing a Non-Existent Device May Return an Unexpected Value
BS78	X	X	No Fix	Some PCIe* CCR Values Are Incorrect
BS79	X	X	No Fix	When in DMI Mode, Port 0's Device_Port_Type Field is Incorrect
BS80	X	X	No Fix	PCIe* TPH Attributes May Result in Unpredictable System Behavior
BS81	X	X	No Fix	Correctable Memory Errors May Result in Unpredictable System Behavior
BS82	X	X	No Fix	Enabling Opportunistic Self-Refresh and Pkg C2 State Can Severely Degrade PCIe Bandwidth
BS83	X	X	No Fix	Mirrored Memory Writes May Lead to System Failures
BS84	X	X	No Fix	IA32_MCi_STATUS ADDRV Bit May be Incorrectly Cleared
BS85	X	X	No Fix	Malformed TLP Power Management Messages May Be Dropped

Errata Number	Steppings		Status	ERRATA
	C-1	C-2		
BS86	X	X	No Fix	Core Frequencies at or Below the DRAM DDR Frequency May Result in Unpredictable System Behavior
BS87	X	X	No Fix	Storage of PEBS Record Delayed Following Execution of MOV SS or STI
BS88	X	X	No Fix	Instruction Fetch May Cause Machine Check if Page Size and Memory Type Was Changed Without Invalidation
BS89	X	X	No Fix	Quad Rank DIMMs May Not be Properly Refreshed During IBT_OFF Mode
BS90	X	X	Fixed	The VT-d Queued Invalidation Status Write May Fail
BS91	X	X	No Fix	Executing The GETSEC Instruction While Throttling May Result in a Processor Hang
BS92	X	X	No Fix	Platform Idle Power Higher May be Higher Than Expected
BS93	X	X	No Fix	PECI Transactions during an S-State Transition May Result in a Platform Cold Reset
BS94	X	X	No Fix	Complex Platform Conditions during a Transition to S4 or S5 State May Result in an Internal Timeout Error
BS95	X	X	No Fix	Performance Monitoring May Overcount Some Events During Debugging
BS96	X	X	No Fix	HDRLOG Registers do not Report the Header for PCIe* Port 1 Packets with Detected Errors
BS97	X	X	No Fix	PECI Temperature Data Values Returned During Reset May be Non-Zero
BS98	X	X	No Fix	PECI Temperature Lower Limit May be as High as 7°C
BS99	X	X	No Fix	TSOD Related SMBus Transactions May Not Complete When Package C-States are Enabled
BS100	X	X	No Fix	The DRAM Power Meter May Not be Accurate
BS101	X	X	No Fix	The Processor Incorrectly Transitions from Polling.Active to Polling.Compliance After Receiving Two TS1 Ordered Sets with the Compliance Bit Set
BS102	X	X	No Fix	Functionally Benign PCIe* Electrical Specification Violation Compendium
BS103	X	X	No Fix	Shallow Self-Refresh Mode is Used During S3
BS104	X	X	No Fix	A Machine Check Exception Due to Instruction Fetch May Be Delivered Before an Instruction Breakpoint
BS105	X	X	No Fix	A PECI RdIAMSR Command Near IERR Assertion May Cause the PECI Interface to Become Unresponsive
BS106	X	X	No Fix	Long Latency Transactions Can Cause I/O Devices on the Same Link to Time Out
BS107	X	X	No Fix	The Coherent Interface Error Code "DA" is Always Flagged
BS108	X	X	No Fix	If Multiple Poison Events Are Detected Within Two Core Clocks, The Overflow Flag May Not be Set
BS109	X	X	No Fix	PCI Express* Capability Structure Not Fully Implemented
BS110	X	X	No Fix	The PCIe* Receiver Lanes Surge Protection Circuit May Intermittently Cause a False Receive Detection on Some PCIe Devices
BS111	X	X	No Fix	Software Reads From LMMIOH_LIMIT Register May be Incorrect
BS112	X	X	No Fix	Intel SpeedStep® Technology May Cause a System Hang
BS113	X	X	No Fix	NTB May Incorrectly Set MSI or MSI-X Interrupt Pending Bits
BS114	X	X	No Fix	Spurious Power Limit Interrupt May Occur at Package C-State Exit
BS115	X	X	No Fix	Spurious Power Limit Interrupt May Occur at Package C-State Exit

Errata Number	Steppings		Status	ERRATA
	C-1	C-2		
BS116	X	X	No Fix	LBR May Contain Incorrect Information When Using FREEZE_LBRS_ON_PMI
BS117	X	X	No Fix	PROCHOT May Be Incorrectly Asserted at Reset
BS118	X	X	No Fix	Programming PDIR And an Additional Precise PerfMon Event May Cause Unexpected PMI or PEBS Events
BS119	X	X	No Fix	FP Data Operand Pointer May Be Incorrectly Calculated After an FP Access Which Wraps a 64-Kbyte Boundary in 16-Bit Code
BS120	X	X	No Fix	Incorrect Address Computed For Last Byte of FXSAVE/FXRSTOR or XSAVE/XRSTOR Image Leads to Partial Memory Update
BS121	X	X	No Fix	FP Data Operand Pointer May Be Incorrectly Calculated After an FP Access Which Wraps a 4-Gbyte Boundary in Code That Uses 32-Bit Address Size in 64-bit Mode
BS122	X	X	No Fix	Execution of VAESIMC or VAESKEYGENASSIST With An Illegal Value for VEX.vvvv May Produce a #NM Exception
BS123	X	X	No Fix	PECI Commands Differing Only in Length Field May be Interpreted as Command Retries
BS124	X	X	No Fix	VM Exits from Real-Address Mode Due to Machine-Check Exceptions May Incorrectly Save RFLAGS.RF as 1
BS125	X	X	No Fix	VM Exits from Real-Address Mode Due to Machine-Check Exceptions May Incorrectly Save RFLAGS.RF as 1
BS126	X	X	No Fix	Rank Sparing May Cause an Extended System Stall
BS127	X	X	No Fix	The Default Value of the More I/O Base Address Field Does Not Comply with the PCI-to-PCI Bridge Architecture Specification
BS128	X	X	No Fix	A Sustained Series of PCIe Posted Upstream Writes Can Lead to Deadlock
BS129	X	X	No Fix	Extraneous Characters Are Included in the Processor Brand String
BS130	X	X	No Fix	IMC Controlled Dynamic DRAM Refresh Rate Can Lead to Unpredictable System Behavior
BS131	X	X	No Fix	Incorrect Error Address Status May Get Logged
BS132	X	X	No Fix	The Machine Check Threshold-Based Error Status Indication May be Incorrect
BS133	X	X	No Fix	IA32_MCi_STATUS Registers May Contain Undefined Data After Reset
BS134	X	X	No Fix	Refresh Cycles for High Capacity DIMMs Are Not Staggered
BS135	X	X	No Fix	A Stream of Snoops Can Lead to a System Hang or Machine Check
BS136	X	X	No Fix	The Value in IA32_MC3_ADDR MSR May Not be Accurate When MCACOD 0119H is Reported in IA32_MC3_Status
BS137	X	X	No Fix	IA32_MCi_STATUS.EN May Not be Set During Certain Machine Check Exceptions
BS138	X	X	No Fix	LLC Cache Correctable Errors Are Not Counted And Logged
BS139	X	X	No Fix	The Processor Incorrectly Transitions From The PCIe* Recovery.RcvrLock LTSSM State to the Configuration.Linkwidth.Start LTSSM State
BS140	X	X	No Fix	Performance Monitor Precise Instruction Retired Event May Present Wrong Indications
BS141	X	X	No Fix	XSAVEOPT May Fail to Save Some State after Transitions Into or Out of STM
BS142	X	X	No Fix	Error Indication in PCIe* Lane Error Status Incorrectly Set When Operating at 8 GT/s
BS143	X	X	No Fix	The Minimum Snoop Latency Requirement That Can be Specified is 64 Microseconds

Errata Number	Steppings		Status	ERRATA
	C-1	C-2		
BS144	X	X	No Fix	A Machine Check May Result in an Unexpected Value in ECX
BS145	X	X	No Fix	System Hang May Occur when Memory Sparing is Enabled
BS146	X	X	No Fix	End Agent PCIe* Packet Errors May Result in a System Hang
BS147	X	X	No Fix	Retraining Cannot be Initiated by Downstream Devices in NTB/NTB or NTB/RP Configurations
BS148	X	X	No Fix	PCIe* Port in NTB Mode Flags Upstream Slot Power Limit Message as UR
BS149	X	X	No Fix	When in DMI Mode, Port 0's Device_Port_Type Field is Incorrect
BS150	X	X	No Fix	PCIe* TPH Attributes May Result in Unpredictable System Behavior
BS151	X	X	No Fix	PCIe* Lane Reversal is Not Supported on All x8 Configurations During REUT Mode
BS152	X	X	No Fix	PCIe* Port 3 Link Training May be Unreliable in NTB Mode
BS153	X	X	No Fix	A Machine-Check Exception Due to Instruction Fetch May Be Delivered Before an Instruction Breakpoint
BS154	X	X	No Fix	Intel® SpeedStep® Technology May Cause a System Hang
BS155	X	X	No Fix	The Accumulated Energy Status Read Service May Report a Power Spike Early in Boot
BS156	X	X	No Fix	Certain Uncorrectable Errors May Cause Loss of PECI Functionality
BS157	X	X	No Fix	Machine Check During VM Exit May Result in VMX Abort
BS158	X	X	No Fix	Routing Intel® High Definition Audio Traffic Through VC1 May Result in System Hang
BS159	X	X	No Fix	Package_Energy_Counter Register May Incorrectly Report Power Consumed by The Execution of Intel® AVX instructions
BS160	X	X	No Fix	Coherent Interface Write Cache May Report False Correctable ECC Errors During Cold Reset
BS161	X	X	No Fix	PCIe* RO May Result in a System Hang or Unpredictable System Behavior
BS162	X	X	No Fix	VT-d Invalidation Time-Out Error May Not be Signaled
BS163	X	X	No Fix	Enhanced Intel SpeedStep® Technology Hardware Coordination Cannot be Disabled
BS164	X	X	No Fix	PCIe* Link Upconfigure Capability is Incorrectly Advertised as Supported
BS165	X	X	No Fix	The IA32_MCi_MISC.HaDbBank Field Should be Ignored
BS166	X	X	No Fix	When a PCIe* x4 Port Detects a Logical Lane 0 Failure, the Link Will Advertise Incorrect Lane Numbers
BS167	X	X	No Fix	Certain PCIe* TLPs May be Dropped
BS168	X	X	No Fix	A Machine Check Exception Concurrent With an I/O SMI May Be Erroneously Reported as Restartable
BS169	X	X	No Fix	VEX.L is Not Ignored with VCVT*2SI Instructions
BS170	X	X	No Fix	The System Agent Temperature is Not Available
BS171	X	X	No Fix	The PCIe* Link at 8.0 GT/s is Transitioning Too Soon to Normal Operation While Training
BS172	X	X	No Fix	An ACM Error May Cause a System Power Down
BS173	X	X	No Fix	Incorrect Retry Packets May Be Sent by a PCIe* x16 Port Operating at 8 GT/s
BS174	X	X	No Fix	The Coherent Interface Error Codes "C2", "C3", "DA" and "DB" are Incorrectly Flagged

Errata Number	Steppings		Status	ERRATA
	C-1	C-2		
BS175	X	X	No Fix	MCI_ADDR May be Incorrect For Cache Parity Errors
BS176	X	X	No Fix	Intel® QuickData DMA Channel Write Abort Errors May Cause a Channel Hang
BS177	X	X	No Fix	The Processor May Not Properly Execute Code Modified Using A Floating-Point Store
BS178	X	X	No Fix	Execution of GETSEC[SEXIT] May Cause a Debug Exception to be Lost
BS179	X	X	No Fix	VM Exits Due to GETSEC May Save an Incorrect Value for “Blocking by STI” in the Context of Probe-Mode Redirection
BS180	X	X	No Fix	Warm Reset May Cause PCIe* Hot-Plug to Fail
BS181	X	X	No Fix	Certain Local Memory Read / Load Retired PerfMon Events May Undercount
BS182	X	X	Plan Fix	Performance Monitor Counters May Produce Incorrect Results
BS183	X	X	No Fix	The Corrected Error Count Overflow Bit in IA32_ MC0_STATUS is Not Updated After a UC Error is Logged
BS184	X	X	No Fix	Spurious Intel® VT-d Interrupts May Occur When the PFO Bit is Set
BS185	X	X	No Fix	Processor May Livelock During On Demand Clock Modulation

Specification Changes

Number	SPECIFICATION CHANGES
	There are no Specification Changes at this time

Specification Clarifications

Number	SPECIFICATION CLARIFICATIONS
	There are no Specification Clarifications at this time.

Documentation Changes

Number	DOCUMENTATION CHANGES
BS1	On-Demand Clock Modulation Feature Clarification

Errata

BS1. An Enabled Debug Breakpoint or Single Step Trap May Be Taken after MOV SS/POP SS Instruction if it is Followed by an Instruction That Signals a Floating Point Exception

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

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

Workaround: As recommended in the *IA32 Intel® Architecture Software Developer's Manual*, the use of MOV SS/POP SS in conjunction with MOV [r/e]SP, [r/e]BP will avoid the failure since the MOV [r/e]SP, [r/e]BP will not generate a floating point exception. Developers of debug tools should be aware of the potential incorrect debug event signaling created by this erratum.

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

BS2. APIC Error "Received Illegal Vector" May be Lost

Problem: APIC (Advanced Programmable Interrupt Controller) may not update the ESR (Error Status Register) 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 Summary Tables of Changes.

BS3. An Uncorrectable Error Logged in IA32_CR_MC2_STATUS May also Result in a System Hang

Problem: Uncorrectable errors logged in IA32_CR_MC2_STATUS MSR (409H) may also result in a system hang causing an Internal Timer Error (MCACOD = 0x0400h) to be logged in another machine check bank (IA32_MCi_STATUS).

Implication: Uncorrectable errors logged in IA32_CR_MC2_STATUS can further cause a system hang and an Internal Timer Error to be logged.

Workaround: None identified.

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

BS4. B0-B3 Bits in DR6 For Non-Enabled Breakpoints May be Incorrectly Set

Problem: Some of the B0-B3 bits (breakpoint conditions detect flags, bits [3:0]) in DR6 may be incorrectly set for non-enabled breakpoints when the following sequence happens:

1. MOV or POP instruction to SS (Stack Segment) selector;

2. Next instruction is FP (Floating Point) that gets FP assist
3. Another instruction after the FP instruction completes successfully
4. A breakpoint occurs due to either a data breakpoint on the preceding instruction or a code breakpoint on the next instruction.

Due to this erratum a non-enabled breakpoint triggered on step 1 or step 2 may be reported in B0-B3 after the breakpoint occurs in step 4.

Implication: Due to this erratum, B0-B3 bits in DR6 may be incorrectly set for non-enabled breakpoints.

Workaround: Software should not execute a floating point instruction directly after a MOV SS or POP SS instruction.

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

BS5. Changing the Memory Type for an In-Use Page Translation May Lead to Memory-Ordering Violations

Problem: Under complex micro architectural conditions, if software changes the memory type for data being actively used and shared by multiple threads without the use of semaphores or barriers, software may see load operations execute out of order.

Implication: Memory ordering may be violated. Intel has not observed this erratum with any commercially available software.

Workaround: Software should ensure pages are not being actively used before requesting their memory type be changed.

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

BS6. Code Segment Limit/Canonical Faults on RSM May be Serviced before Higher Priority Interrupts/Exceptions and May Push the Wrong Address Onto the Stack

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

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

Workaround: None identified.

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

BS7. Corruption of CS Segment Register During RSM While Transitioning From Real Mode to Protected Mode

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

Implication: The corruption of the bottom two bits of the CS segment register will have no impact unless software explicitly examines the CS segment register between enabling protected mode and the first far JMP. Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3A: System Programming Guide, Part 1, in the section

titled "Switching to Protected Mode" recommends the far JMP immediately follows the write to CR0 to enable protected mode. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

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

BS8. Debug Exception Flags DR6.B0-B3 Flags May be Incorrect for Disabled Breakpoints

Problem: When a debug exception is signaled on a load that crosses cache lines with data forwarded from a store and whose corresponding breakpoint enable flags are disabled (DR7.G0-G3 and DR7.L0-L3), the DR6.B0-B3 flags may be incorrect.

Implication: The debug exception DR6.B0-B3 flags may be incorrect for the load if the corresponding breakpoint enable flag in DR7 is disabled.

Workaround: None identified.

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

BS9. DR6 May Contain Incorrect Information When the First Instruction After a MOV SS,r/m or POP SS is a Store

Problem: Normally, each instruction clears the changes in DR6 (Debug Status Register) caused by the previous instruction. However, the instruction following a MOV SS,r/m (MOV to the stack segment selector) or POP SS (POP stack segment selector) instruction will not clear the changes in DR6 because data breakpoints are not taken immediately after a MOV SS,r/m or POP SS instruction. Due to this erratum, any DR6 changes caused by a MOV SS,r/m or POP SS instruction may be cleared if the following instruction is a store.

Implication: When this erratum occurs, incorrect information may exist in DR6. This erratum will not be observed under normal usage of the MOV SS,r/m or POP SS instructions (i.e., following them with an instruction that writes [e/r]SP). When debugging or when developing debuggers, this behavior should be noted.

Workaround: None identified.

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

BS10. EFLAGS Discrepancy on Page Faults and on EPT-Induced VM Exits after a Translation Change

Problem: 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 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 Summary Tables of Changes.

BS11. Fault on ENTER Instruction May Result in Unexpected Values on Stack Frame

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

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

Workaround: None identified.

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

BS12. Faulting MMX Instruction May Incorrectly Update x87 FPU Tag Word

Problem: Under a specific set of conditions, MMX stores (MOVD, MOVQ, MOVNTQ, MASKMOVQ) which cause memory access faults (#GP, #SS, #PF, or #AC), may incorrectly update the x87 FPU tag word register.

This erratum will occur when the following additional conditions are also met.

- The MMX store instruction must be the first MMX instruction to operate on x87 FPU state (i.e. the x87 FP tag word is not already set to 0x0000).
- For MOVD, MOVQ, MOVNTQ stores, the instruction must use an addressing mode that uses an index register (this condition does not apply to MASKMOVQ).

Implication: If the erratum conditions are met, the x87 FPU tag word register may be incorrectly set to a 0x0000 value when it should not have been modified.

Workaround: None identified

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

BS13. FREEZE_WHILE_SMM Does Not Prevent Event From Pending PEBS During SMM

Problem: In general, a 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 SMM (System Management Mode). Due to this erratum, if

1. A performance counter overflowed before an 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 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 Summary Tables of Changes.

BS14. General Protection Fault (#GP) for Instructions Greater than 15 Bytes May be Preempted

Problem: When the processor encounters an instruction that is greater than 15 bytes in length, a #GP is signaled when the instruction is decoded. Under some circumstances, the #GP fault may be preempted by another lower priority fault (e.g. Page Fault (#PF)). However, if the preempting lower priority faults are resolved by the operating system and the instruction retried, a #GP fault will occur.

Implication: Software may observe a lower-priority fault occurring before or in lieu of a #GP fault. Instructions of greater than 15 bytes in length can only occur if redundant prefixes are placed before the instruction.

Workaround: None identified.

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

BS15. #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, see the Summary Tables of Changes.

BS16. An Event May Intervene Before a System Management Interrupt That Results from IN or INS

Problem: If an I/O instruction (IN, INS, OUT, or OUTS) results in an SMI (system-management interrupt), the processor will set the IO_SMI bit at offset 7FA4H in SMRAM. This interrupt should be delivered immediately after execution of the I/O instruction so that the software handling the SMI can cause the I/O instruction to be re-executed. Due to this erratum, it is possible for another event (e.g., a non maskable interrupt) to be delivered before the SMI that follows the execution of an IN or INS instruction.

Implication: If software handling an affected SMI uses I/O instruction restart, the handler for the intervening event will not be executed.

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

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

BS17. IRET under Certain Conditions May Cause an Unexpected Alignment Check Exception

Problem: In IA-32e mode, it is possible to get an Alignment Check Exception (#AC) on the IRET instruction even though alignment checks were disabled at the start of the IRET. This can only occur if the IRET instruction is returning from CPL3 code to CPL3 code. IRETs from CPL0/1/2 are not affected. This erratum can occur if the EFLAGS value on the stack has the AC flag set, and the interrupt handler's stack is misaligned. In IA-32e mode, RSP is aligned to a 16-byte boundary before pushing the stack frame.

Implication: In IA-32e mode, under the conditions given above, an IRET can get a #AC even if alignment checks are disabled at the start of the IRET. This erratum can only be observed with a software generated stack frame.

Workaround: Software should not generate misaligned stack frames for use with IRET.

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

BS18. LER MSRs May Be Unreliable

Problem: Due to certain internal processor events, updates to the LER (Last Exception Record) 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 Summary Tables of Changes.

BS19. 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 LBR (Last Branch Record), BTS (Branch Trace Store) and BTM (Branch Trace Message) 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 Summary Tables of Changes.

BS20. 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 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 Summary Tables of Changes.

BS21. 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 Summary Tables of Changes.

BS22. MOV To/From Debug Registers Causes Debug Exception

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

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

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

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

BS23. PEBS Record not Updated when in Probe Mode

Problem: When a performance monitoring counter is configured for PEBS (Precise Event Based Sampling), overflows of the counter can result in storage of a PEBS record in the PEBS buffer. Due to this erratum, if the overflow occurs during probe mode, it may be ignored and a new PEBS record may not be added to the PEBS buffer.

Implication: Due to this erratum, the PEBS buffer may not be updated by overflows that occur during probe mode.

Workaround: None identified.

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

BS24. Performance Monitor Counter INST_RETIRED.STORES May Count Higher than Expected

Problem: Performance Monitoring counter INST_RETIRED.STORES (Event: C0H) is used to track retired instructions which contain a store operation. Due to this erratum, the processor may also count other types of instructions including WRMSR and MFENCE.

Implication: Performance Monitoring counter INST_RETIRED.STORES may report counts higher than expected.

Workaround: None identified.

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

BS25. Performance Monitoring Event FP_MMX_TRANS_TO_MMX May Not Count Some Transitions

Problem: Performance Monitor Event FP_MMX_TRANS_TO_MMX (Event CCH, Umask 01H) counts transitions from x87 Floating Point (FP) to MMX™ instructions. Due to this erratum, if only a small number of MMX instructions (including EMMS) are executed immediately after the last FP instruction, a FP to MMX transition may not be counted.

Implication: The count value for Performance Monitoring Event FP_MMX_TRANS_TO_MMX may be lower than expected. The degree of undercounting is dependent on the occurrences of the erratum condition while the counter is active. Intel has not observed this erratum with any commercially available software.

Workaround: None Identified.

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

BS26. REP MOVS/STOS Executing with Fast Strings Enabled and Crossing Page Boundaries with Inconsistent Memory Types may use an Incorrect Data Size or Lead to Memory-Ordering Violations.

Problem: Under certain conditions as described in the Software Developers Manual section "Out-of-Order Stores For String Operations in Pentium 4, Intel Xeon, and P6 Family Processors" the processor performs REP MOVS or REP STOS as fast strings. Due to this erratum fast string REP MOVS/REP STOS instructions that cross page boundaries from WB/WC memory types to UC/WP/WT memory types, may start using an incorrect data size or may observe memory ordering violations.

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

- UC the data size of each write will now always be 8 bytes, as opposed to the original data size.
- WP the data size of each write will now always be 8 bytes, as opposed to the original data size and there may be a memory ordering violation.
- WT there may be a memory ordering violation.

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

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

BS27. Reported Memory Type May Not Be Used to Access the VMCS and Referenced Data Structures

Problem: Bits 53:50 of the IA32_VMX_BASIC MSR report the memory type that the processor uses to access the VMCS and data structures referenced by pointers in the VMCS. Due to this erratum, a VMX access to the VMCS or referenced data structures will instead use the memory type that the MTRRs (memory-type range registers) specify for the physical address of the access.

Implication: Bits 53:50 of the IA32_VMX_BASIC MSR report that the WB (write-back) memory type will be used but the processor may use a different memory type.

Workaround: Software should ensure that the VMCS and referenced data structures are located at physical addresses that are mapped to WB memory type by the MTRRs.

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

BS28. Single Step Interrupts with Floating Point Exception Pending May Be Mishandled

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

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

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

Workaround: None identified.

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

BS29. The Processor May Report a #TS Instead of a #GP Fault

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

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

Workaround: None identified.

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

BS30. VM Exits Due to "NMI-Window Exiting" May Be Delayed by One Instruction

Problem: If VM entry is executed with the "NMI-window exiting" VM-execution control set to 1, a VM exit with exit reason "NMI window" should occur before execution of any instruction if there is no virtual-NMI blocking, no blocking of events by MOV SS, and no blocking of events by STI. If VM entry is made with no virtual-NMI blocking but with blocking of events by either MOV SS or STI, such a VM exit should occur after execution of one

instruction in VMX non-root operation. Due to this erratum, the VM exit may be delayed by one additional instruction.

Implication: VMM software using "NMI-window exiting" for NMI virtualization should generally be unaffected, as the erratum causes at most a one-instruction delay in the injection of a virtual NMI, which is virtually asynchronous. The erratum may affect VMMs relying on deterministic delivery of the affected VM exits.

Workaround: None identified.

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

BS31. Values for LBR/BTS/BTM Will be Incorrect after an Exit from SMM

Problem: After a return from SMM (System Management Mode), the CPU will incorrectly update the LBR (Last Branch Record) and the BTS (Branch Trace Store), hence rendering their data invalid. The corresponding data if sent out as a BTM on the system bus will also be incorrect.

Note: This issue would only occur when one of the 3 above mentioned debug support facilities are used.

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

Workaround: None identified.

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

BS32. VPHMINPOSUW Instruction in VEX Format Does Not Signal #UD (Invalid Opcode Exception) When vex.vvvv !=1111

Problem: Processor does not signal #UD fault when executing the reserved instruction VPHMINPOSUW with vex.vvvv!=1111. The VPHMINPOSUW instruction is described in greater detail in the Intel® Advanced Vector Extensions Programming Reference.

Implication: Executing VPHMINPOSUW with vex.vvvv != 1111 results in same behavior as vex.vvvv= 1111.

Workaround: SW should not use VPHMINPOSUW with vex.vvvv != 1111 in order to ensure future compatibility.

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

BS33. 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 Summary Tables of Changes.

BS34. VMREAD/VMWRITE Instruction May Not Fail When Accessing an Unsupported Field in VMCS

Problem: The Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2B states that execution of VMREAD or VMWRITE should fail if the value of the instruction's register source operand corresponds to an unsupported field in the VMCS (Virtual Machine Control Structure). The correct operation is that the logical processor will set the ZF (Zero Flag), write 0CH into the VM-instruction error field and for VMREAD leave the instruction's destination operand unmodified. Due to this erratum, the instruction

may instead clear the ZF, leave the VM-instruction error field unmodified and for VMREAD modify the contents of its destination operand.

Implication: Accessing an unsupported field in VMCS will fail to properly report an error. In addition, VMREAD from an unsupported VMCS field may unexpectedly change its destination operand. Intel has not observed this erratum with any commercially available software.

Workaround: Software should avoid accessing unsupported fields in a VMCS.

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

BS35. Unexpected #UD on VZEROALL/VZEROUPPER

Problem: Execution of the VZEROALL or VZEROUPPER instructions in 64-bit mode with VEX.W set to 1 may erroneously cause a #UD (invalid-opcode exception).

Implication: The affected instructions may produce unexpected invalid-opcode exceptions in 64-bit mode.

Workaround: Compilers should encode VEX.W = 0 for the VZEROALL and VZEROUPPER instructions.

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

BS36. Execution of Opcode 9BH with the VEX Opcode Extension May Produce a #NM Exception

Problem: Attempt to use opcode 9BH with a VEX opcode extension should produce a #UD (Invalid-Opcode) exception. Due to this erratum, if CR0.MP and CR0.TS are both 1, the processor may produce a #NM (Device-Not-Available) exception if one of the following conditions exists:

- 66H, F2H, F3H or REX as a preceding prefix;
- An illegal map specified in the VEX.mmmmm field;

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

Workaround: Software should not use opcode 9BH with the VEX opcode extension.

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

BS37. Enabling Opportunistic Self-Refresh and Pkg C2 State Can Severely Degrade PCIe* Bandwidth

Problem: Due to this erratum, enabling opportunistic self-refresh can lead to the memory controller over-aggressively transitioning DRAM to self-refresh mode when the processor is in Pkg C2 state.

Implication: The PCIe interface peak bandwidth can be degraded by as much as 90%.

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

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

BS38. An Unexpected Page Fault or EPT Violation May Occur After Another Logical Processor Creates a Valid Translation for a Page

Problem: An unexpected page fault (#PF) or EPT violation may occur for a page under the following conditions:

- The paging structures initially specify no valid translation for the page.
- Software on one logical processor modifies the paging structures so that there is a valid translation for the page (e.g., by setting to 1 the present bit in one of the paging-structure entries used to translate the page).
- Software on another logical processor observes this modification (e.g., by accessing a linear address on the page or by reading the modified paging-structure entry and seeing value 1 for the present bit).

- Shortly thereafter, software on that other logical processor performs a store to a linear address on the page.

In this case, the store may cause a page fault or EPT violation that indicates that there is no translation for the page (e.g., with bit 0 clear in the page-fault error code, indicating that the fault was caused by a not-present page). Intel has not observed this erratum with any commercially available software.

Implication: An unexpected page fault may be reported. There are no other side effects due to this erratum.

Workaround: System software can be constructed to tolerate these unexpected page faults. See Section "Propagation of Paging-Structure Changes to Multiple Processors" of Volume 3B of IA-32 Intel® Architecture Software Developer's Manual, for recommendations for software treatment of asynchronous paging-structure updates.

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

BS39. Faulting Executions of XRSTOR May Update State Inconsistently

Problem: The state updated by a faulting XRSTOR instruction may vary from one execution to another.

Implication: Software that relies on x87/SSE/AVX state following a faulting execution of XRSTOR may behave inconsistently.

Workaround: Software handling a fault on an execution of XRSTOR can compensate for execution variability by correcting the cause of the fault and executing XRSTOR again.

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

BS40. 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 a #UD (Invalid-Opcode) 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 Summary Tables of Changes.

BS41. Unexpected #UD on VPEXTRD/VPINSRD

Problem: Execution of the VPEXTRD or VPINSRD instructions outside of 64-bit mode with VEX.W set to 1 may erroneously cause a #UD (invalid-opcode exception).

Implication: The affected instructions may produce unexpected invalid-opcode exceptions outside 64-bit mode.

Workaround: Software should encode VEX.W = 0 for executions of the VPEXTRD and VPINSRD instructions outside 64-bit mode.

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

BS42. #GP May be Signaled When Invalid VEX Prefix Precedes Conditional Branch Instructions

Problem: When a 2-byte opcode of a conditional branch (opcodes 0F8xH, for any value of x) instruction resides in 16-bit code-segment and is associated with invalid VEX prefix, it may sometimes signal a #GP fault (illegal instruction length > 15-bytes) instead of a #UD (illegal opcode) fault.

Implication: Due to this erratum, #GP fault instead of a #UD may be signaled on an illegal instruction.

Workaround: None identified.

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

BS43. LBR, BTM or BTS Records May have Incorrect Branch From Information After an EIST/T-state/S-state/C1E Transition or Adaptive Thermal Throttling

Problem: The "From" address associated with the LBR (Last Branch Record), BTM (Branch Trace Message) or BTS (Branch Trace Store) may be incorrect for the first branch after a transition of:

- EIST (Enhanced Intel® SpeedStep Technology)
- T-state (Thermal Monitor states)
- S1-state (ACPI package sleep state)
- C1E (Enhanced C1 Low Power state)
- Adaptive Thermal Throttling

Implication: When the LBRs, BTM or BTS are enabled, some records may have incorrect branch "From" addresses for the first branch after a transition of EIST, T-states, S-states, C1E, or Adaptive Thermal Throttling.

Workaround: None identified.

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

BS44. A Write to the IA32_FIXED_CTR1 MSR May Result in Incorrect Value in Certain Conditions

Problem: Under specific internal conditions, if software tries to write the IA32_FIXED_CTR1 MSR (30AH) a value that has all bits [31:1] set while the counter was just about to overflow when the write is attempted (i.e. its value was 0xFFFF FFFF FFFF), then due to this erratum the new value in the MSR may be corrupted.

Implication: Due to this erratum, IA32_FIXED_CTR1 MSR may be written with a corrupted value.

Workaround: Software may avoid this erratum by writing zeros to the IA32_FIXED_CTR1 MSR, before the desired write operation.

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

BS45. L1 Data Cache Errors May be Logged With Level Set to 1 Instead of 0

Problem: When an L1 Data Cache error is logged in IA32_MCi_STATUS[15:0], which is the MCA Error Code Field, with a cache error type of the format 0000 0001 RRRR TTLL, the LL field may be incorrectly encoded as 01b instead of 00b.

Implication: An error in the L1 Data Cache may report the same LL value as the L2 Cache. Software should not assume that an LL value of 01b is the L2 Cache.

Workaround: None identified.

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

BS46. PECI RdPkgConfig() May Return Invalid Data For an Unsupported Channel

Problem: The processor's PECI facility can report the current temperature of each of the DIMMs on a specified channel (PECI RdPkgConfig command, index 14H, DIMM_Temperature_Read). Valid channel numbers range from 0 to 3. Channel numbers outside of the valid range should be detected and flagged. Due to this erratum, meaningless values are returned without an error flag when 4 is specified as the channel number.

Implication: Using channel 4 with the PECI RdPkgConfig DIMM_Temperature_Read command does not return an error flag.

Workaround: None identified.

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

BS47. VM Entries That Return From SMM Using VMLAUNCH May Not Update The Launch State of the VMCS

Problem: Successful VM entries using the VMLAUNCH instruction should set the launch state of the VMCS to "launched". Due to this erratum, such a VM entry may not update the launch state of the current VMCS if the VM entry is returning from SMM.

Implication: Subsequent VM entries using the VMRESUME instruction with this VMCS will fail. RFLAGS.ZF is set to 1 and the value 5 (indicating VMRESUME with non-launched VMCS) is stored in the VM-instruction error field. This erratum applies only if dual monitor treatment of SMI and SMM is active.

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

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

BS48. Interrupt From Local APIC Timer May Not Be Detectable While Being Delivered

Problem: If the local-APIC timer's CCR (current-count register) 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 LVT timer register and then reading the bit in the IRR (interrupt-request register) 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 DCR (Divide Configuration Register) 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 Summary Tables of Changes.

BS49. End Agent PCIe* Packet Errors May Result in a System Hang

Problem: PCIe agents are required by the PCIe Base Specification to identify and report packet errors. Due to this erratum, certain invalid completion types from the end agent are not correctly handled by the processor.

Implication: If a PCIe end agent issues certain invalid completion types, the system may hang.

Workaround: None identified.

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

BS50. Poison Packets Will be Reported to PCIe* Port 1a When Forwarded to Port 1b

Problem: With respect to data poisoning, the processor IIO module supports forwarding poisoned information between the coherent interface and PCIe and vice-versa. Also the processor IIO module supports forwarding poisoned data between peer PCIe ports. When the PCIe Ports 1a and 1b are configured as x4, the outbound Poison Error is reported on Port 1a when a poison packet is forwarded to Port 1b.

Implication: When Ports 1a and 1b are configured as x4 ports, Poison Errors reported on the root port are unreliable.

Workaround: None identified.

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

BS51. IA32_MCi_ADDR Overwritten in The Case of Multiple Recoverable Instruction Fetch Errors

Problem: The instruction fetch machine check error (MCACOD 0x150) is a SRAR (Software Recoverable Action Required) error. The address of the location with the error is provided in the corresponding IA32_MCi_ADDR MSR. When multiple instruction fetch errors are logged as part of a single machine check event, as indicated by setting of the Overflow (bit 62) in the IA32_MCi_STATUS MSR, then recovery is not possible. Due to this erratum, when multiple instruction fetch errors are logged in the same bank, the IA32_MCi_MISC MSR contains all of the correct information including the proper setting for Overflow (bit 62); however, the IA32_MCi_ADDR MSR is overwritten with a value that corresponds to neither the first or second error.

Implication: When debugging failures associated with the instruction fetch machine check error and the Overflow bit is set, the value in IA32_MCi_ADDR will not be valid.

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

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

BS52. PCIe* Link May Not Train to Full Width

Problem: During PCIe link training, the receiver looks at symbols in the TS1 and TS2 Ordered Sets as indicators of lane polarity inversion. If polarity inversion is detected, the receiver must invert those lane(s). Due to this erratum, the receiver may incorrectly set polarity inversion.

Implication: PCIe links may not train to full width.

Workaround: None identified. Perform a Secondary Bus Reset on the link up to three times to achieve full width.

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

BS53. Spurious SMIs May Occur Due to MEMHOT# Assertion

Problem: The IMC (Integrated Memory Controller) can be programmed to generate an SMI (System Management Interrupt) on an internal MEMHOT# event assertion through the MHOT_SMI_EN field (MH_MAINCNTL Bus: 1; Device: 15; Function: 0; Offset: 104H; bit[17]) or on assertion of the external MEMHOT[1:0]#pin though the MHOT_EXT_SMI_EN field (MH_MAINCNTL Bus: 1; Device: 15; Function: 0; Offset: 104H; bit[18]). Due to this erratum, a spurious SMI may be generated every 500uS if both internal and external MEMHOT events are enabled simultaneously.

Implication: Due to this erratum, excessive SMI generation may occur.

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

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

BS54. The PCIe* Current Compensation Value Default is Incorrect

Problem: The default current compensation values for PCIe buffers may result in non-optimal performance.

Implication: The PCIe buffers will not perform as well as possible and performance could be compromised.

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

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

BS55. The PCIe* Link at 8.0 GT/s is Transitioning Too Soon to Normal Operation While Training

Problem: The PCIe bus uses high speed serial links that must go through a training process to allow both transmitter and receiver to make adjustments in behavior to optimize the

signaling between the transmitter and receiver. When a PCIe compliant device must train or retrain the link, training sequences are used. The device must allow enough time for the training to complete before transitioning to normal operation. In the case of PCIe equalization at 8.0 GT/s the processor is not allowing enough time to optimize signaling before attempting normal operation.

Implication: Due to this erratum, unexpected system behavior may be observed.

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

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

BS56. A First Level Data Cache Parity Error May Result in Unexpected Behavior

Problem: When a load occurs to a first level data cache line resulting in a parity error in close proximity to other software accesses to the same cache line and other locked accesses the processor may exhibit unexpected behavior.

Implication: Due to this erratum unpredictable system behavior may occur. Intel has not observed this erratum with any commercially available system.

Workaround: None identified.

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

BS57. PECI Write Requests That Require a Retry Will Always Time Out

Problem: PECI 3.0 introduces a 'Host Identification' field as a way for the PECI host device to identify itself to the PECI client. This is intended for use in future PECI systems that may support more than one PECI originator. Since PECI 3.0 systems do not support the use of multiple originators, PECI 3.0 host devices should zero out the unused Host ID field. PECI 3.0 also introduces a 'retry' bit as a way for the PECI host to indicate to the client that the current request is a 'retry' of a previous read or write operation. Unless the PECI 3.0 host device zeroes out the byte containing the 'Host ID & Retry bit' information, PECI write requests that require a retry will never complete successfully.

Implication: PECI write requests that require a retry may never complete successfully. Instead, they will return a timeout completion code of 81H for a period ranging from 1ms to 30ms if the 'RETRY' bit is asserted.

Workaround: PECI 3.0 host devices should zero out the byte that contains the Host ID and Retry bit information for all PECI requests at all times including retries.

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

BS58. The Vswing of the PCIe* Transmitter Exceeds The Specification

Problem: The PCIe Specification defines a limit for the Vswing (Voltage Swing) of the differential lines that make up a lane to be 1200 mV peak-to-peak when operating at 2.5 GT/s and 5 GT/s. Intel has found that the processor's PCIe transmitter may exceed this specification. Peak-to-peak swings on a limited number of samples have been observed up to 1450 mV.

Implication: For those taking direct measurements of the PCIe transmit traffic coming from the processor may detect that the Vswing exceeds the PCIe Specification. Intel has not observed any functional failures due to this erratum.

Workaround: None identified.

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

BS59. When a Link is Degraded on a Port due to PCIe* Signaling Issues Correctable Receiver Errors May be Reported on The Neighboring Port

Problem: PCI Express interface incorporates a recovery mechanism when certain link degradation occurs by retraining the link without impacting the pending transactions. When a link is degraded on a specific port due to PCIe signaling issues, it is possible

that correctable receiver errors are reported on the neighboring (logically adjacent) port. The correctable receiver errors are indicated by the PCIe AER Correctable error bit (XPGLBERRSTS CPUBUS(0); Device 0-3; Function 0-3; Offset 230H; Bit 2).

Implication: Software that logs errors on the PCIe interface must be aware that errors detected on a specific port could be due to either an error on that specific port or on a neighboring port.

Workaround: None identified.

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

BS60. A CMCI is Only Generated When the Memory Controller's Correctable Error Count Threshold is Exceeded

Problem: A CMCI (corrected machine-check error interrupt) should be generated when the number of corrected errors for a bank reaches the corrected error threshold programmed into the IA32_MCi_CTL2 bits [14:0]. For memory scrubbing errors, IA32_MCi_STATUS.MCACOD (bits [15:0]) with value of 000x_0000_1100_xxxx (where x stands for zero or one), a CMCI will not be generated until the number of errors has exceeded the threshold in IA32_MCi_CTL2 by 1.

Implication: The CMCI will not be generated when expected but rather will be generated on the next corrected error for the bank.

Workaround: It is possible for BIOS to contain a workaround for this issue. It should be noted that with this workaround if the threshold is programmed to a value of 0, a read of the value will return 1 and the threshold will be 1. All other valid threshold values for the bank will be read back correctly and function as expected.

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

BS61. PCIe* Rx DC Common Mode Impedance is Not Meeting the Specification

Problem: When the PCIe Rx termination is not powered, the DC Common Mode impedance has the following requirement: ≥10 kΩ over 0-200 mV range with respect to ground and ≥20 kΩ for voltages ≥200 mV with respect to ground. The processor's PCIe Rx do not meet this requirement at 85 degrees C or greater. In a limited number of samples Intel has measured an impedance as low as 9.85 kΩ at 50mV.

Implication: Intel has not observed any functional impact due to this violation with any commercially available system.

Workaround: None identified.

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

BS62. A Modification to the Multiple Message Enable Field Does Not Affect The AER Interrupt Message Number field

Problem: The (Advanced Error Interrupt) Message Number field (RPERRSTS Devices 0-3; Functions 0-3; Offset 178H; bits[31:27]) should be updated when the number of messages allocated to the root port is changed by writing the Multiple Message Enable field (MSIMSGCTL Device 3; Function 0; Offset 62H; bits[6:4]). However, writing the Multiple Message Enable in the root port does not update the Advanced Error Interrupt Message Number field.

Implication: Due to this erratum, software can allocate only one MSI (Message Signaled Interrupt) to the root port.

Workaround: None identified.

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

BS63. Unexpected PCIe* Set_Slot_Power_Limit Message on Writes to LNKCON

Problem: The processor sends the PCIe Set_Slot_Power_Limit message on writes to the Slot Capabilities (SLTCAP Devices 0-3; Functions 0-3; Offset A4H) register. Due to this erratum, the processor also sends PCIe the Set_Slot_Power_Limit message on writes to the LNKCON (CPUBUS(0); Devices 0-3; Functions 0-3; Offset A0H) register.

Implication: For those monitoring the PCIe* traffic going across the link, the unexpected PCIe Set_Slot_Power_Limit Message will be detected whenever a write to the LNKCON register occurs. Intel has not observed any functional failures due to this erratum on any commercially available system.

Workaround: None identified.

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

BS64. PCIe* Link Bandwidth Notification Capability is Incorrect

Problem: A value of 1 in the Link_Bandwidth_Notification_Capability field (LKNCAP bit 21) for a PCIe device indicates support for the Link Bandwidth Notification status and interrupt mechanisms. Due to this erratum, this field for ports 2c, 2d, 3c and 3d (LKNCAP Bus 0; Device 2,3; Function 2,3; Offset 09Ch; bit 21) always reads as 0 when it should read as 1.

Implication: Software that reads this field for the listed ports will incorrectly conclude that the Link Bandwidth Notification status and interrupt mechanisms are not available.

Workaround: Software should ignore the value of the Link_Bandwidth_Notification_Capability field for ports 2c, 2d, 3c, and 3d.

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

BS65. Locked Accesses Spanning Cachelines That Include PCI Space May Lead to a System Hang

Problem: A locked memory access which splits across a cacheline boundary that suffers a master abort on a PCI bus may lead to a system hang.

Implication: Aborted split lock accesses may cause PCI devices to become inoperable until a platform reset. Intel has not observed this erratum with commercially available software.

Workaround: None identified.

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

BS66. Cold Boot May Fail Due to Internal Timer Error

Problem: The processors may not complete a cold boot (i.e. a boot from a power-off state) due to an internal timer error machine check, IA32_MCi_STATUS.MCACOD of 0000_0100_0000_0000. This will result in the processor asserting IERR (Internal Error).

Implication: The processor may signal IERR during a cold boot when the system is initializing.

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

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

BS67. PCIe* Rx Common Mode Return Loss is Not Meeting The Specification

Problem: The PCIe Specification requires that the Rx Common Mode Return Loss in the range of 0.05 to 2.5 GHz must be limited to -6 dB. The processor’s PCIe Rx do not meet this requirement. The PCIe Rx Common Mode Return at 500MHz has been found to be between -3.5 and -4 dB on a limited number of samples.

Implication: Intel has not observed any functional failures due to this erratum with any commercially available PCIe devices.

Workaround: None identified.

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

BS68. The Most Significant Bit of the CEC Cannot be Cleared Once Set

Problem: The most significant bit of the CEC (Corrected Error Count IA32_MCi_STATUS (i=12-19), bit 52) cannot be cleared once it has been set.

Implication: In the case that software attempts to clear the CEC and the count exceeds 3FFFH, software will read incorrect CEC values on subsequent accesses and additional CMCIs (Corrected Machine Check Error Interrupts) will not be generated.

Workaround: None identified. Software can avoid this erratum by setting corrected error threshold to a value less than 3FFFH, enable CMCI and clearing the error count before it exceeds 3FFFH.

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

BS69. PCIe* Adaptive Equalization May Not Train to the Optimal Settings

Problem: In the case of the PCIe equalization procedure for 8 GT/s, the Downstream Port's (e.g. the processor's) TXEQ (transmitter equalization settings) can be fine tuned for each Lane during a process called Adaptive Equalization Phase 3. Due to this erratum, the processor may not direct the end-agent to the optimal TXEQ settings.

Implication: The PCIe link may not be as robust as possible potentially leading to a higher bit error rate than expected.

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

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

BS70. A Core May Not Complete Transactions to The Caching Agent When C-States Are Enabled Leading to an Internal Timer Error

Problem: When multiple cores have outstanding transactions targeted to a single caching agent and one of the cores enters a Core C-state before completing the transaction with the targeted caching agent an internal timer machine check error may occur (IA32_MCi_STATUS.MCACOD of 0000_0100_0000_0000).

Implication: Due to this erratum, the processor may experience an internal timer error.

Workaround: None identified.

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

BS71. TSC is Not Affected by Warm Reset

Problem: The TSC (Time Stamp Counter MSR 10H) should be cleared on reset. Due to this erratum the TSC is not affected by warm reset.

Implication: The TSC is not cleared by a warm reset. The TSC is cleared by power-on reset as expected. Intel has not observed any functional failures due to this erratum.

Workaround: None identified.

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

BS72. Warm Resets May be Converted to Power-on Resets When Recovering From an IERR

Problem: When a warm reset is attempted and an IERR (Internal Error) happens as indicated by the IA32_MCi_STATUS.MCACOD of 0000_0100_0000_0000, a power-on reset occurs instead.

Implication: The values in the machine check bank will be lost as a result of the power-on reset. This prevents a OS, BIOS or the BMC (Baseboard Management Controller) from logging the content of the error registers or taking any post-reset actions that are dependent on the machine check information.

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

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

BS73. Port 3a Capability_Pointer Field is Incorrect When Configured in PCIe* Mode

Problem: The Capability_Pointer field (CAPPTR Bus 0; Device 3; Function 0; Offset 34H; bits [7:0]) should have its value based on the configured mode of the port, PCIe or NTB (Non-Transparent Bridge). Due to this erratum, this field reports the NTB value (60H) when in PCIe mode instead of the PCIe value (40H).

Implication: Software depending on the value of this field may not behave as expected.

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

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

BS74. Processor May not Restore the VR12 DDR3 Voltage Regulator Phases upon Pkg C3 State Exit

Problem: During the Pkg (Package) C3 state entry, the processor directs the VR12 DDR3 voltage regulators to shed phases to reduce power consumption. Due to this erratum, the processor may not restore all VR12 DDR3 voltage regulator phases upon Pkg C3 state exit. The VR12 DDR3 voltage regulators require all phases to keep the DDR3 voltage plane in tolerance for proper memory subsystem functioning during normal system operation.

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 Summary Tables of Changes.

BS75. The Equalization Phase Successful Bits Are Not Compliant to The PCIe* Specification

Problem: PCIe Specification states that if the Phase 1 of Transmitter Equalization completes successfully as indicated by the LNKSTS2.Equalization Phase 1 Successful (Devices 0-3; Functions 0-3; bit[2]) bit being set to one and if the Phase 2 and 3 link training phases are bypassed, the LNKSTS2.Equalization Phase 3 Successful (Devices 0-3; Functions 0-3; bit[4]) and LNKSTS2.Equalization Phase 2 Successful (bit[3]) bits should be set to one. Due to this erratum, the processor will only set the Equalization Phase 2 or 3 Successful bits if the phases are completed successfully.

Implication: Due to this erratum, Equalization Phase 2 and 3 Successful bits may not be set. Intel has not observed any functional failure with commercially available PCIe devices.

Workaround: None identified.

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

BS76. Four Outstanding PCIe* Configuration Retries May Cause Deadlock

Problem: PCIe configuration retries are allowed for older generation PCI/PCI-X bridges that take a long time to respond to configuration cycles after a reset. Due to this erratum, a fifth configuration cycle following the fourth PCIe configuration retry may not make progress, resulting in a deadlock.

Implication: A deadlock could occur. Intel has not observed this erratum with any commercially available system.

Workaround: When configuring devices on PCI/PCI-X buses, BIOS should wait for configuration cycles to complete before issuing subsequent configuration cycles.

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

BS77. A PECI RdPciConfigLocal Command Referencing a Non-Existent Device May Return an Unexpected Value

Problem: Configuration reads to non-existent PCI configuration registers should return 0FFFF_FFFFH. Due to this erratum, when the PECI RdPciConfigLocal command references a non-existent PCI configuration register, the value 0000_0000H may be returned instead of the expected 0FFFF_FFFFH.

Implication: A PECI RdPciConfigLocal command referencing a non-existent device may observe a return value of 0000_0000H. Software expecting a return value of 0FFFF_FFFFH to identify non-existent devices may not work as expected.

Workaround: Software that performs enumeration via the PECI "RdPciConfigLocal" command should interpret 0FFFF_FFFFH and 0000_0000H values for the Vendor Identification and Device Identification Register as indicating a non-existent device.

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

BS78. Some PCIe* CCR Values Are Incorrect

Problem: The CCR (Class Code Register) value for the following devices should be 088000H instead is 000000H:

- Bus 0; Device 6; Function 1-7; Offset 09H; bits [23:0]
- Bus 0; Device 7; Function 0-4; Offset 09H; bits [23:0]

Implication: Due to this erratum, the CCR base and sub-class status of the listed PCIe devices is incorrectly reported and may cause software to conclude that these devices are host bridges and are not general system peripherals.

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

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

BS79. When in DMI Mode, Port 0's Device_Port_Type Field is Incorrect

Problem: When in DMI mode, the Device_Port_Type field (PXPCAP Bus 0; Device 0; Function 0; Offset 92H; bits [7:4]) should read as 9H (DMI mode) but incorrectly reads as 4H (PCIe* mode).

Implication: Software may incorrectly conclude that this port is operating in PCIe mode when it is actually being used in the DMI mode.

Workaround: None identified.

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

BS80. PCIe* TPH Attributes May Result in Unpredictable System Behavior

Problem: TPH (Transactions Processing Hints) are optional aids to optimize internal processing of PCIe transactions. Due to this erratum, certain transactions with TPH attributes may be misdirected, resulting in unpredictable system behavior.

Implication: Use of the TPH feature may affect system stability.

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

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

BS81. Correctable Memory Errors May Result in Unpredictable System Behavior

Problem: Under certain conditions, the processor may not detect or correct a correctable memory error.

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

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

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

BS82. Enabling Opportunistic Self-Refresh and Pkg C2 State Can Severely Degrade PCIe Bandwidth

Problem: Due to this erratum, enabling opportunistic self-refresh can lead to the memory controller over-aggressively transitioning DRAM to self-refresh mode when the processor is in Pkg C2 state.

Implication: The PCIe* interface peak bandwidth can be degraded by as much as 90%.

Workaround: A BIOS workaround has been identified. Please refer to the latest version of the BIOS Spec Update and release notes.

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

BS83. Mirrored Memory Writes May Lead to System Failures

Problem: In mirrored memory mode, each channel manages its memory write bandwidth resources. Due to this erratum, if a channel in mirrored memory mode is heavily utilized, it is possible for issued writes to exceed available bandwidth resulting in write failures.

Implication: A system hang or unpredictable system behavior may occur.

Workaround: A BIOS workaround has been identified. Please refer to BIOS Specification Update, Intel® Romley Platform CPU/QPI/Memory Reference Code version 1.0.006 or later and release notes.

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

BS84. IA32_MCi_STATUS ADDRV Bit May be Incorrectly Cleared

Problem: The IA32_MCi_STATUS MSR's ADDRV bit (bit 58) is set upon logging an error in order to indicate that the contents of the IA32_MCi_ADDR MSR is valid. Due to this erratum, a cancelled speculative load of poisoned data spanning a cacheline boundary can clear the ADDRV flag associated with a previously logged error report.

Implication: The clearing of the ADDRV flag in IA32_MCi_STATUS when this erratum occurs will result in the loss of the address logged in a correctable error report. It should be noted that a cancelled speculative load of poisoned data that crosses a cacheline boundary is an unusual occurrence.

Workaround: None identified.

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

BS85. Malformed TLP Power Management Messages May Be Dropped

Problem: The PCIe* Base Specification requires Power Management Messages to use the default Traffic Class designator, TC0, and receivers to check for violations of this rule. Due to this erratum, a TLP using a non-default Traffic Class designator will be dropped, rather than handled as a Malformed TLP.

Implication: An Advanced Error Reporting ERR_FATAL notification will not be logged for Malformed TLP Power Management Messages.

Workaround: None identified

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

BS86. Core Frequencies at or Below the DRAM DDR Frequency May Result in Unpredictable System Behavior

Problem: The Intel® SpeedStep® Technology can dynamically adjust the core operating frequency to as low as 1200 MHz. Due to this erratum, under complex conditions and when the cores are operating at or below the DRAM DDR frequency, unpredictable system behavior may result.

Implication: Systems using Intel SpeedStep Technology with DDR3-1333 or DDR3-1600 memory devices are subject to unpredictable system behavior.

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

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

BS87. Storage of PEBS Record Delayed Following Execution of MOV SS or STI

Problem: When a performance monitoring counter is configured for PEBS (Precise Event Based Sampling), overflow of the counter results in storage of a PEBS record in the PEBS buffer. The information in the PEBS record represents the state of the next instruction to be executed following the counter overflow. Due to this erratum, if the counter overflow occurs after execution of either MOV SS or STI, storage of the PEBS record is delayed by one instruction.

Implication: When this erratum occurs, software may observe storage of the PEBS record being delayed by one instruction following execution of MOV SS or STI. The state information in the PEBS record will also reflect the one instruction delay.

Workaround: None identified.

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

BS88. 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 (e.g., 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 Summary Tables of Changes.

BS89. Quad Rank DIMMs May Not be Properly Refreshed During IBT_OFF Mode

Problem: The Integrated Memory Controller incorporates a power savings mode known as IBT_OFF (Input Buffer Termination disabled). Due to this erratum, Quad Rank DIMMs may not be properly refreshed during IBT_OFF mode.

Implication: Use of IBT_OFF mode with Quad Rank DIMMs may result in unpredictable system behavior.

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

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

BS90. The VT-d Queued Invalidation Status Write May Fail

Problem: Intel® Virtualization Technology for Directed I/O (Intel® VT-d) queued invalidation operations issue a status write to modify a semaphore. Due to this erratum, the status write may fail.

Implication: When using queued invalidation operations, a failed status write can result in unpredictable system behavior.

Workaround: If operating without queued invalidations, interrupt re-mapping, and X2APIC features is feasible, then VT-d invalidations should be performed using the VT-d register facility (c.f., VTD0_CTXCMD [offset 028h], VTD1_CTXCMD [offset 1028h], VTD0_INVADDRREG [offset 0200h] and VTD0_IOTLBINV [offset 0208h], VTD1_INVADDRREG [offset 1200h] and VTD1_IOTLBINV [offset 1208h] in the VT-d register region with a base address specified through the VTBAR register at 0:5:0, offset 0180h). If those operational limitations are not feasible, disable VT-d through BIOS facilities. This will prevent the use of Intel VT-d, including X2APIC and TXT facilities that are dependent on Intel VT-d.

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

BS91. Executing The GETSEC Instruction While Throttling May Result in a Processor Hang

Problem: If the processor throttles due to either high temperature thermal conditions or due to an explicit operating system throttling request (TT1) while executing GETSEC[SENTER] or GETSEC[SEXIT] instructions, then under certain circumstances, the processor may hang. Intel has not been observed this erratum with any commercially available software.

Implication: Possible hang during execution of GETSEC instruction.

Workaround: None Identified.

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

BS92. Platform Idle Power Higher May be Higher Than Expected

Problem: The processor may not place the associated DRAM subsystem in the lowest allowed power state during Package C3 and Package C6 states. This may cause the platform idle power to be higher than expected.

Implication: Platform average power and idle power may be higher than expected.

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

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

BS93. PECI Transactions during an S-State Transition May Result in a Platform Cold Reset

Problem: Due to this erratum, a PECI transaction during an S-state transition may result in an unexpected platform cold reset rather than an S-state transition.

Implication: Use of PECI transactions during an S-state transition can result in a platform reset that terminates transitioning to the desired S-state.

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

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

BS94. Complex Platform Conditions during a Transition to S4 or S5 State May Result in an Internal Timeout Error

Problem: Due to this erratum, the BIOS sequencing associated with S4 (sometimes known as “Hibernate”) and S5 (also known as “Soft Off”), when undertaken with certain complex platform conditions, can result in an internal timeout error as indicated by IA32_MCi_STATUS.MCACOD of 0000_0100_0000_0000 and IERR assertion. This internal timeout error stops the platform S-state sequencing before platform power down occurs. Certain platforms may have logic that, upon detection of the failure to reach power down, initiates a cold reset sequence.

Implication: S4 state or S5 state may not be reliably entered; the platform may not reach the very low power condition.

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

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

BS95. Performance Monitoring May Overcount Some Events During Debugging

Problem: If the debug-control register (DR7) is configured so that some but not all of the breakpoints in the debug-address registers (DR0-DR3) are enabled and one or more of the following performance-monitoring counters are locally enabled (via IA32_CR_PERMON_EVNTSEL_CNTR{3:0}):

BR_INST_RETIRED
BR_MISP_RETIRED
FP_ASSIST
FP_ASSIST
INST_RETIRED
MACHINE_CLEARS
MEM_LOAD_UOPS_LLC_HIT_RETIRED
MEM_LOAD_UOPS_MISC_RETIRED.LLC_MISS
MEM_LOAD_UOPS_RETIRED
MEM_TRANS_RETIRED
MEM_UOPS_RETIRED
OTHER_ASSISTS
ROB_MISC_EVENTS.LBR_INSERTS
UOPS_RETIRED

Any of the globally enabled (via IA32_CR_EMON_PERF_GLOBAL_CTRL) counters may overcount certain events when a disabled breakpoint condition is met.

Implication: Performance-monitor counters may indicate a number greater than the number of events that occurred.

Workaround: Software can disable all breakpoints by clearing DR7. Alternatively, software can ensure that, for a breakpoint disabled in DR7, the corresponding debug-address register contains an address that prevents the breakpoint condition from being met (e.g., a non-canonical address).

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

BS96. HDRLOG Registers do not Report the Header for PCIe* Port 1 Packets with Detected Errors

Problem: The HDRLOG registers contain the header information of the first PCIe packet detected that contains errors. Because of this erratum, the Port 1 (IOU2) HDRLOG registers (CPUBUS(0), Device 1, Function 0; Offsets 164H, 168H, 16CH, 170H) do not reflect the header of a packet with a detected error.

Implication: The HDRLOG registers cannot be used to debug the receipt of packets with detected errors on Port 1.

Workaround: None identified.

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

BS97. PECI Temperature Data Values Returned During Reset May be Non-Zero

Problem: The processor PECI power-up time line presented in the Intel® Core™ i7 Processor Family for the LGA-2011 Socket Datasheet - Volume 1 & Volume 2 defines the value returned by the PECI GetTemp() command as 0x0000 – the maximum value – during the ‘Data Not Ready’ (DNR) phase (starting approximately 100 µS after PWRGOOD assertion and lasting until approximately 500 µS after RESET de-assertion). Due to this erratum, the GetTemp() command returns a small negative number during the DNR phase.

Implication: The temperature reported during the PECI DNR phase may be below the maximum and therefore may not have the intended effect of causing platform fans to operate at full speed until the actual processor temperature becomes available.

Workaround: Processor thermal management solutions utilizing PECI should operate platform fans at full speed during the PECI DNR phase.

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

BS98. PECI Temperature Lower Limit May be as High as 7°C

Problem: PECI reports temperatures as an offset from the PROCHOT threshold (a negative value when the temperature is below the PROCHOT threshold, zero when at or above that threshold). If the temperature is below 0°C, PECI responds with an "Invalid Temperature" encoding (8002H). Due to this erratum, PECI may indicate an invalid temperature when the actual temperature is as high as 7°C.

Implication: An invalid temperature report from PECI indicates the actual temperature is 7°C or lower. Platform facilities depending PECI to provide accurate temperature readings between 0°C and 7°C may not function correctly.

Workaround: None identified.

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

BS99. TSOD Related SMBus Transactions May Not Complete When Package C-States are Enabled

Problem: The processor may not complete SMBus (System Management Bus) transactions targeting the TSOD (Temperature Sensor On DIMM) when Package C-States are enabled. Due to this erratum, if the processor transitions into a Package C-State while an SMBus transaction with the TSOD is in process, the processor will suspend receipt of the transaction. The transaction completes while the processor is in a Package C-State. Upon exiting Package C-State, the processor will attempt to resume the SMBus transaction, detect a protocol violation, and log an error.

Implication: When Package C-States are enabled, the SMBus communication error rate between the processor and the TSOD may be higher than expected.

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

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

BS100. The DRAM Power Meter May Not be Accurate

Problem: The DRAM Power Meter uses VR (Voltage Regulator) current readings in combination with weighted activity counters to provide a running estimate of DRAM subsystem power. Due to this erratum, the DRAM Power Meter may not be sufficiently accurate for system power management purposes.

Implication: The DRAM Power Meter cannot be relied upon to provide accurate DRAM subsystem power measurements. Reduced or variable system performance may be a side effect.

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

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

BS101. The Processor Incorrectly Transitions from Polling.Active to Polling.Compliance After Receiving Two TS1 Ordered Sets with the Compliance Bit Set

Problem: The processor PCIe* interface incorrectly transitions from the Polling.Active Link state to the Polling.Compliance Link state after receiving two TS1 Ordered Sets with the Compliance Bit set instead of the eight TS1 Ordered Sets required by the specification.

Implication: It is possible that the PCIe link may enter Polling.Compliance Link state unexpectedly. Exposure to this erratum requires bit errors on the Compliance Receive bit (Byte 5, Bit 4) on sequential TS1 ordered sets.

Workaround: None identified.

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

BS102. Functionally Benign PCIe* Electrical Specification Violation Compendium

Problem: Violations of PCIe electrical specifications listed in the table below have been observed.

Specification	Violation Description
De-emphasis ratio limit: -3.5±0.5 dB	Ave: -3.8 dB, Min: -4.09 dB
At 5 GT/s operation, the receiver must tolerate AC common mode voltage of 300 mV (Peak-to-Peak) and must tolerate 78.1 ps jitter	Simultaneous worst case AC common mode voltage and worst case jitter during 5 GT/s operation may result in intermittent failures leading to subsequent recovery events
TTX-UPW-TJ (uncorrelated total pulse width jitter) maximum of 24ps	Samples have measured as high as 25ps
The Transmitter PLL bandwidth and peaking for PCIe at 5 GT/s is either 8-16 MHz with 3 dB of peaking or 5-16 MHz with 1 dB of peaking	Samples have measured 7.8-16 MHz with 1.3 dB of peaking
During the LTSSM Receiver Detect State, common-mode resistance to ground is 40-60 ohms.	Samples have measured up to 100 ohms.
8 GT/s Receiver Stressed Eye	Samples marginally pass or fail the 10^{-12} BER target under stressed eye conditions
8 GT/s PLL Bandwidth: 2 to 4 MHz with 2 dB peaking.	Samples have a measured bandwidth of up to 4.1 MHz

Implication: Intel has not observed failures from the violations listed in this erratum on any commercially available platforms and/or using commercially available PCIe devices.

Workaround: None identified.

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

BS103. Shallow Self-Refresh Mode is Used During S3

Problem: The processor should be instructing DRAM to utilize deep self-refresh at entry into the S3 state. Due to this erratum, the processor is instructing the DRAM to use shallow self-refresh upon entry into the S3 state.

Implication: The power dissipation of the DRAMs will be greater than expected during S3 state.

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

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

BS104. A Machine Check Exception Due to Instruction Fetch May Be Delivered Before an Instruction Breakpoint

Problem: Debug exceptions due to instruction breakpoints take priority over exceptions resulting from fetching an instruction. Due to this erratum, a machine check exception resulting from the fetch of an instruction may take priority over an instruction breakpoint if the instruction crosses a 32-byte boundary and the second part of the instruction is in a 32-byte poisoned instruction fetch block.

Implication: Instruction breakpoints may not operate as expected in the presence of a poisoned instruction fetch block.

Workaround: None identified.

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

BS105. A PECI RdIAMSR Command Near IERR Assertion May Cause the PECI Interface to Become Unresponsive

Problem: When a PECI RdIAMSR command is issued to the processor near the time that the processor is experiencing an internal timeout error, as indicated by IA32_MCi_STATUS.MCACOD of 0000_0100_0000_0000 and IERR assertion, the PECI interface may issue an 81H (timeout) response. After a timeout response, the processor will ignore future PECI commands until it is reset.

Implication: Due to this erratum, PECI commands typically used to debug a processor that is not behaving normally – RdPkgConfig and RdPciConfig – may not be available after an internal time out error.

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

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

BS106. Long Latency Transactions Can Cause I/O Devices on the Same Link to Time Out

Problem: Certain long latency transactions – e.g., master aborts on inbound traffic, locked transactions, peer-to-peer transactions, or vendor defined messages – conveyed over the PCIe* and DMI2 interfaces can block the progress of subsequent transactions for extended periods. In certain cases, these delays may lead to I/O device timeout that can result in device error reports and/or device off-lining.

Implication: Due to this erratum, devices that generate PCIe or DMI2 traffic characterized by long latencies can interfere with other traffic types on the same link. This may result in reduced I/O performance and device timeout errors. USB traffic can be particularly sensitive to these delays.

Workaround: Avoid the contributing conditions. This can be accomplished by separating traffic types to be conveyed on different links and/or reducing or eliminating long latency transactions.

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

BS107. The Coherent Interface Error Code "DA" is Always Flagged

Problem: The Coherent Interface Error Status Registers (IRPP0ERRST and IRPP1ERRST at CPUBUS(0), Device 5, Function 2, Offsets 230H and 2B0H respectively) indicate that an error has been detected by the Coherent Interface. Bit 13 of the IRPP0ERRST and IRPP1ERRST registers indicate that a Protocol Queue/Table Overflow or Underflow (DA) error has occurred. Due to this erratum, the processor always logs the DA error flag.

Implication: The DA error flag is indeterminate.

Workaround: Mask off the DA error flag (bit 13) of the IRPP0ERRCTL and IRPP1ERRCTL registers at CPUBUS(0), Device 5, Function 2, Offsets 234H and 2B4H respectively.

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

BS108. If Multiple Poison Events Are Detected Within Two Core Clocks, The Overflow Flag May Not be Set

Problem: If multiple poison events are detected within two core clocks, the error is logged with an IA32_MCi_STATUS.MCACOD of 0000_0001_0011_0100 but the IA32_MCi_STATUS.OVER (bit [60]) may not be set.

Implication: Due to this erratum, only one poison event may be reported by a logical processor when more than one poison event was encountered.

Workaround: None identified.

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

BS109. PCI Express* Capability Structure Not Fully Implemented

Problem: According to the PCIe Base Specification, "The PCI Express Capability structure is required for all PCI Express device functions". Due to this erratum, some PCI Express Capabilities Fields were not implemented ("Device Capability", "Device Status" and "Device Control") for CPUBUS[0], Device 5, Function 2, reads to these fields will return zero.

Implication: Software that depends on the PCI Express Capability Structure fields Device Capability, Device Status and/or Device Control will not operate properly.

Workaround: None identified.

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

BS110. The PCIe* Receiver Lanes Surge Protection Circuit May Intermittently Cause a False Receive Detection on Some PCIe Devices

Problem: The processor implements a surge protection circuit on the PCIe receiver lanes. Due to this erratum, during platform power-on some PCIe devices may trigger the surge protection circuit causing a false receive detect. If this unexpected detection occurs before the processor's PCIe lane termination impedances are enabled and the resulting PCIe device link training enters the link training Polling.Active state, the PCIe device may incorrectly transition into the Polling.Compliance state.

Implication: After platform power-on, some PCIe devices may not exit from the compliance state causing the link to fail to train or the link may train to a degraded width.

Workaround: It is possible for BIOS to contain a workaround for this erratum. Please refer to memory reference code version 0.8.301 or later with release notes, the latest version of the Intel® Server Platform Services Release (SPS_02.01.05.012.0 or later) with release notes, and the latest version of the Intel® Management Engine Firmware 7.1 Release 1 (7.1.20.1128 or later) with release notes.

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

BS111. Software Reads From LMMIOH_LIMIT Register May be Incorrect

Problem: The MMIOH is a memory-mapped I/O region relocatable above 4 GB. Due to this erratum, software reads of the LMMIOH_LIMIT register (Local MMIO High Base, Device: 5, Function: 0, Offset 118H) may yield incorrect results, although software writes to this register function as expected.

Implication: Software depending on LMMIOH_LIMIT register reads may not behave as expected. Intel has not identified any commercially available software that is affected by the erratum.

Workaround: None identified.

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

BS112. Intel SpeedStep® Technology May Cause a System Hang

Problem: Intel SpeedStep® Technology dynamically changes core operating frequencies. Due to this erratum, under complex conditions, core frequency changes may result in a system hang.

Implication: Intel SpeedStep Technology may cause a system hang.

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

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

BS113. NTB May Incorrectly Set MSI or MSI-X Interrupt Pending Bits

Problem: The NTB (Non-transparent Bridge) may incorrectly set MSI (Message Signaled Interrupt) pending bits in MSIPENDING (BAR PB01BASE,SB01BASE; Offset 74H) while operating in MSI-X mode or set MSI-X pending bits in PMSIXPBA (BAR PB01BASE, SB01BASE; Offset 03000H) while operating in MSI mode.

Implication: Due to this erratum, NTB incorrectly sets MSI or MSI-X pending bits. The correct pending bits are also set and it is safe to ignore the incorrectly set bits.

Workaround: None Identified.

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

BS114. Spurious Power Limit Interrupt May Occur at Package C-State Exit

Problem: The processor monitors power consumption and uses that information to limit core operating frequency. Due to this erratum, power consumption may be improperly calculated by the processor during Package C-states. As a result, the processor may incorrectly signal a power limit interrupt.

Implication: In response to a power limit interrupt, the OS may choose to operate the processor at its minimum frequency for several milliseconds after the Package C-state exit.

Workaround: None identified. The OS can mask these interrupts by setting the Power Limit Interrupt Enable field (bit 24) in the IA32_THERM_INTERRUPT MSR (19BH) to 0.

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

BS115. Using I/O Peer-to-Peer Write Traffic Across an NTB May Lead to a Hang

Problem: If two systems are connected via an NTB (Non-Transparent Bridge), either the internal NTB or an external NTB, and both systems attempt to send I/O peer-to-peer write traffic across the NTB either to memory or an I/O device on the remote system, it is possible for both systems to deadlock.

Implication: Due to this erratum, using I/O peer-to-peer write traffic across an NTB may lead to a hang.

Workaround: A BIOS workaround has been identified. Please refer to the latest version of the BIOS spec update and release notes. However, the work-around could lead to periods of low performance due to starvation of PCIe traffic as it allows the arbiter to grant access to another device if the current granted device is blocked by resource limits of its intended target, rather than wait until the current winner has sent at least one transaction.

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

BS116. LBR May Contain Incorrect Information When Using FREEZE_LBRS_ON_PMI

Problem: When FREEZE_LBRS_ON_PMI is enabled (bit 11 of IA32_DEBUGCTL MSR (1D9H) is set), and a taken branch retires at the same time that a PMI (Performance Monitor Interrupt) occurs, then under certain internal conditions the record at the top of the LBR stack may contain an incorrect "From" address.

Implication: When the LBRs are enabled with FREEZE_LRBS_ON_PMI, the "From" address at the top of the LBR stack may be incorrect.

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

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

BS117. PROCHOT May Be Incorrectly Asserted at Reset

Problem: The PROCHOT signal is used to indicate elevated processor temperatures during normal operation and is used for FRB (Fault Resilient Boot) actions during the reset sequence. Due to this erratum, the elevated temperature indication usage of PROCHOT can persist into reset and subsequently can cause improper FRB actions.

Implication: Elevated die temperatures at reset time may impair platform operation.

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

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

BS118. Programming PDIR And an Additional Precise PerfMon Event May Cause Unexpected PMI or PEBS Events

Problem: PDIR (Precise Distribution for Instructions Retired) mechanism is activated by programming INST_RETIRED.ALL (event C0H, umask value 00H) on Counter 1. When PDIR is activated in PEBS (Precise Event Based Sampling) mode with an additional precise PerfMon event, an incorrect PMI or PEBS event may occur.

Implication: Due to this erratum, when another PEBS event is programmed along with PDIR, an incorrect PMI or PEBS event may occur.

Workaround: Software should not program another PEBS event in conjunction with the PDIR mechanism.

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

BS119. FP Data Operand Pointer May Be Incorrectly Calculated After an FP Access Which Wraps a 64-Kbyte Boundary in 16-Bit Code

Problem: The FP (Floating Point) Data Operand Pointer is the effective address of the operand associated with the last non-control FP instruction executed by the processor. If an 80-bit FP access (load or store) occurs in a 16-bit mode other than protected mode (in which case the access will produce a segment limit violation), the memory access wraps a 64-Kbyte boundary, and the FP environment is subsequently saved, the value contained in the FP Data Operand Pointer may be incorrect.

Implication: Due to this erratum, the FP Data Operand Pointer may be incorrect. Wrapping an 80-bit FP load around a segment boundary in this way is not a normal programming practice. Intel has not observed this erratum with any commercially available software.

Workaround: If the FP Data Operand Pointer is used in an operating system which may run 16-bit FP code, care must be taken to ensure that no 80-bit FP accesses are wrapped around a 64-Kbyte boundary.

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

BS120. Incorrect Address Computed For Last Byte of FXSAVE/FXRSTOR or XSAVE/XRSTOR Image Leads to Partial Memory Update

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

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

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

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

BS121. FP Data Operand Pointer May Be Incorrectly Calculated After an FP Access Which Wraps a 4-Gbyte Boundary in Code That Uses 32-Bit Address Size in 64-bit Mode

Problem: The FP (Floating Point) Data Operand Pointer is the effective address of the operand associated with the last non-control FP instruction executed by the processor. If an 80-bit FP access (load or store) uses a 32-bit address size in 64-bit mode and the memory access wraps a 4-Gbyte boundary and the FP environment is subsequently saved, the value contained in the FP Data Operand Pointer may be incorrect.

Implication: Due to this erratum, the FP Data Operand Pointer may be incorrect. Wrapping an 80-bit FP load around a 4-Gbyte boundary in this way is not a normal programming practice. Intel has not observed this erratum with any commercially available software.

Workaround: If the FP Data Operand Pointer is used in a 64-bit operating system which may run code accessing 32-bit addresses, care must be taken to ensure that no 80-bit FP accesses are wrapped around a 4-Gbyte boundary.

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

BS122. 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, see the Summary Tables of Changes.

BS123. PECI Commands Differing Only in Length Field May be Interpreted as Command Retries

Problem: Due to this erratum, the processor interprets any PECI read or write command that accesses the processor, a downstream PCI device, or package configuration space and differs from the preceding request only in the length field as a retry request. That is, a retry will be inferred by the processor even if the read length and write length fields don’t match between two consecutive requests, regardless of the state of the host retry bit on the succeeding request.

Implication: Back-to-back PECI commands that are identical with the exception of the length field may yield incorrect results if processor retry completion codes are ignored by the PECI host.

Workaround: PECI hosts should retry timed-out commands until they complete successfully by reissuing a PECI command sequence identical to the originally timed-out command.

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

BS124. VM Exits from Real-Address Mode Due to Machine-Check Exceptions May Incorrectly Save RFLAGS.RF as 1

Problem: If a machine check is encountered while fetching an instruction, and if the resulting machine-check exception causes a VM exit, the VM exit should save an RFLAGS value in the guest-state area of the VMCS with the RF value that existed at the time of the machine check. Due to this erratum, such VM exits that occur in real-address mode may save RFLAGS.RF as 1 even if it had been 0.

Implication: The processor may fail to report an instruction breakpoint following a return to real-address mode via VM entry.

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

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

BS125. Rank Sparing May Cause an Extended System Stall

Problem: The Integrated Memory Controller sequencing during a rank sparing copy operation blocks all writes to the memory region associated with the rank being taken out of service. Due to this erratum, this block can result in a system stall that persists until the sparing copy operation completes.

Implication: The system can stall at unpredictable times which may be observed as one time instance of system unavailability.

Workaround: A BIOS workaround has been identified. Please refer to Intel® Romley Platform CPU/QPI/Memory Reference Code version 1.0.006 or later and release notes.

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

BS126. The Integrated Memory Controller Does Not Enforce CKE High for tXSDLL DCLKs After Self-Refresh

Problem: The JEDEC STANDARD DDR3 SDRAM Specification (No. 79-3E) requires that the CKE signal be held high for tXSDLL DCLKs after exiting self-refresh before issuing commands that require a locked DLL (Delay-Locked Loop). Due to this erratum, the Integrated Memory Controller may not meet this requirement with 512Mb, 1Gb, and 2Gb devices in single rank per channel configurations.

Implication: Violating tXSDLL may result in DIMM clocking issues and may lead to unpredictable system behavior.

Workaround: A BIOS workaround has been identified. Please refer to the Intel® Romley Platform CPU/QPI/Memory Reference Code(RC), version 0.8.0 or later.

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

BS127. The Default Value of the More I/O Base Address Field Does Not Comply with the PCI-to-PCI Bridge Architecture Specification

Problem: The PCI-to-PCI Bridge Architecture Specification defines the default value of the More I/O Base Address Field (IOBAS CPUBUS(0); Device 0-3; Function 0-3; Offset 1Ch; bits [3:2]) to 0. Due to this erratum, the processor's default value is 3.

Implication: It is possible that system software will generate an error due to this erratum.

Workaround: A BIOS workaround has been identified. Please refer to the latest version of the Intel® Xeon® Processor E5-1600/2400/2600/4600 Product Families BIOS Specification Update.

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

BS128. A Sustained Series of PCIe Posted Upstream Writes Can Lead to Deadlock

Problem: Due to this erratum, a sustained series of PCIe posted upstream writes to the same cache line, with no other access of that same cache line, may cause a deadlock.

Implication: Under a complex set of conditions, a sustained series of PCIe posted upstream writes targeting the same cache line can lead to deadlock. Intel has not been observed this erratum with any commercially available system.

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

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

BS129. Extraneous Characters Are Included in the Processor Brand String

Problem: The Processor Brand String is provided by the CPUID instruction for leaf values EAX=80000002H, 80000003H, and 80000004H. Each execution of the three CPUID leaf value returns 16 ASCII bytes of the Processor Brand String in the EAX, EBX, ECX, and EDX registers. Due to this erratum, an extra zero character ("0", 30H ASCII code) and space character (" ", 20H ASCII code) are inserted after the processor number in the brand string output. In the following example brand string, the extraneous characters are underlined: "Intel® Xeon® CPU E5-2680 0 @ 2.70 GHz".

Implication: An extraneous "0" and "space" character are included in the Processor Brand String.

Workaround: The extraneous characters may be ignored or removed by software.

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

BS130. IMC Controlled Dynamic DRAM Refresh Rate Can Lead to Unpredictable System Behavior

Problem: DRAMs require a 2x refresh rate when operating above 85°C. Due to this erratum, the IMC (Integrated Memory Controller) logic intended to double the refresh rate when

DRAM temperature exceeds 85°C can cause DRAM access failures, leading to unpredictable system behavior.

Implication: The IMC is not able to dynamically adjust the DRAM refresh rate based on DRAM temperature. If DRAMs may be operated above 85°C then BIOS must configure the IMC for a doubled refresh rate.

Workaround: A BIOS workaround has been identified. Please refer to the Intel® Romley Platform CPU/QPI/Memory Reference Code (RC), Version 0.9.000 or later and release notes.

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

BS131. Incorrect Error Address Status May Get Logged

Problem: When a correctable Machine Check event with a valid address precedes an uncorrectable Machine Check event without a valid address, the IA32_MCi_STATUS OVER flag (bit 62) should be set and ADDRV flag (bit 58) should be cleared. Due to this erratum, both flags may be set.

Implication: The Machine Check report logged may incorrectly indicate valid address information when the OVER flag is set.

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

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

BS132. The Machine Check Threshold-Based Error Status Indication May be Incorrect

Problem: A corrected cache hierarchy data or tag error is reported in IA32_MCi_STATUS.MCACOD (bits [15:0]) with value of 000x_0001_xxxx_xx01 (where x stands for zero or one). An error status indication (bits [54:53]) value of 10B indicates that the corrected error count has exceeded the yellow threshold. Due to this erratum, subsequent corrections after the yellow indication has been set may change the error status indication to green (bits [54:53] equal to 00B).

Implication: The threshold-based error status indication is unreliable.

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

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

BS133. IA32_MCi_STATUS Registers May Contain Undefined Data After Reset

Problem: Due to this erratum, if the RESET_N signal is asserted while the processor is in a Package C State the IA32_MCi_STATUS registers may contain undefined data after the processor completes the reset. In particular, the IA32_MCi_STATUS.VAL (bit[63]) may be set incorrectly indicating a valid Machine Check has been logged.

Implication: Invalid errors may be reported in the IA32_MCi_STATUS registers.

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

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

BS134. Refresh Cycles for High Capacity DIMMs Are Not Staggered

Problem: Certain high capacity DIMMs, typically Quad Rank RDIMMs and LR-DIMMs, may exceed instantaneous and short-term power limits if refresh cycles are not correctly staggered. Due to this erratum, the Integrated Memory Controller is unable to stagger refresh cycles.

Implication: Some DIMMs may exceed power limits during refresh operations leading 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 Summary Tables of Changes.

BS135. A Stream of Snoops Can Lead to a System Hang or Machine Check

Problem: Due to this erratum, a stream of snoop requests to a single cache slice may cause the processor in that slice to livelock, resulting in a system hang or Internal Timer Error machine check indicated by IA32_MCI_STATUS.MCACOD (bits 15:0, 0000 0100 0000 0000).

Implication: A system hang or machine check may occur. 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, see the Summary Tables of Changes.

BS136. The Value in IA32_MC3_ADDR MSR May Not be Accurate When MCACOD 0119H is Reported in IA32_MC3_Status

Problem: Under certain conditions, when the The Machine Check Error Code (MCACOD) in the IA32_MC3_STATUS (MSR 040DH) register is 0119H, the value in IA32_MC3_ADDR MSR (40EH) may refer to the incoming MLC (Mid-Level Cache) cache line instead of the evicted cache line.

Implication: The address in IA32_MC3_ADDR MSR (40EH) may not be accurate for MLC cache read errors with MSCOD of 119H.

Workaround: None identified.

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

BS137. IA32_MCi_STATUS.EN May Not be Set During Certain Machine Check Exceptions

Problem: Due to this erratum, IA32_MCi_STATUS.EN may not be set as expected after the MLC (Mid-Level Cache) has logged a fatal error with a MCACOD value of 000X_0001_XXXX_XX10 (where X stands for zero or one) and signaled an MCE (Machine Check Error) as a result of encountering poisoned data.

Implication: The value of IA32_MCi_STATUS.EN may be inconsistent with signaling an MCE while logging a fatal error, however a machine check exception is still signaled.

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

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

BS138. LLC Cache Correctable Errors Are Not Counted And Logged

Problem: LLC Cache correctable errors are logged in the Corrected_Error_Count field bits [53:38] of the IA32_MC[19:12]_STATUS MSR. Due to this erratum, LLC Cache corrections are not counted and logged.

Implication: Software using the corrected error count may not function correctly. A CMCI (corrected machine check error interrupt) may not be generated when the error threshold programmed in IA32_CR_MC[19:12]_CTL2.ERROR_THRESHOLD (bits [14:0]) would otherwise be expected to be met.

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

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

BS139. The Processor Incorrectly Transitions From The PCIe* Recovery.RcvrLock LTSSM State to the Configuration.Linkwidth.Start LTSSM State

Problem: When a PCIe link is operating at 2.5 GT/s and the processor's LTSSM (Link Training and Status State Machine) is in Recovery.RcvrLock state, the processor expects to receive TS1 ordered sets within 24 ms. If it does not receive the TS1s in the allotted time, the LTSSM should transition to the Detect state. Due to this erratum, if the processor does not receive TS1s within 24ms, it will transition to Configuration.LinkWidth.Start. In that

state, if it receives no TS1s, it will transition to Detect. If it receives TS1s, it will configure the link appropriately and return to L0.

Implication: The state transition sequence from the Recovery.RcvrLock LTSSM state to the Configuration.Linkwidth.Start LTSSM state is in violation of the PCIe Specification. Intel has not observed any functional failures due to this erratum with any commercially available PCIe devices.

Workaround: None identified.

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

BS140. Performance Monitor Precise Instruction Retired Event May Present Wrong Indications

Problem: When the PDIR (Precise Distribution for Instructions Retired) 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/or an invalid counter state.

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

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

BS141. XSAVEOPT May Fail to Save Some State after Transitions Into or Out of STM

Problem: The XSAVEOPT instruction may optimize performance by not saving state that has not been modified since the last execution of XRSTOR. This optimization should occur only if the executions of XSAVEOPT and XRSTOR are either both or neither in SMM (system-management mode). Due to this erratum, this optimization may be performed by the first execution of XSAVEOPT after a transition into or out of the STM (SMM-transfer monitor) if the most recent execution of XRSTOR occurred before that transition. For transitions into the STM, the erratum applies only to transitions using the VMCALL instruction. This erratum can occur only if the two executions are at the same privilege level, use the same linear address, and are either both or neither in VMX non-root operation. The erratum does not apply if software in SMM never uses XRSTOR or XSAVEOPT.

Implication: This erratum may lead to unpredictable system behavior.

Workaround: STM software should execute the XRSTOR instruction with the value 0 in EDX:EAX after each transition into the STM (after setting CR4.OSXSAVE) and before each transition out of the STM. Bytes 512 to 575 of the save area used by XRSTOR should be allocated in memory, but bytes 0 to 511 need not be. Bytes 512 to 535 should all be 0.

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

BS142. Error Indication in PCIe* Lane Error Status Incorrectly Set When Operating at 8 GT/s

Problem: The Lane Error Status field in bits[15:0] of LNERRSTS (Device 1; Function 0,1; Offset 258H; and Device 2,3; Function 0,1,2,3; Offset 258H) is used to monitor errors on the PCIe lanes. Due to this erratum, the LNERRSTS bits associated with the lanes operating at 8 GT/s port are spuriously set.

Implication: LNERRSTS cannot be used to reliably monitor errors on the PCIe lanes operating at 8 GT/s.

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

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

BS143. The Minimum Snoop Latency Requirement That Can be Specified is 64 Microseconds

Problem: The PCIE_ILTR_OVRD CSR (Device 10; Function 1; Offset 78H) and SW_LTR_OVRD MSR (0A02H) include fields defined to allow specification of a required maximum snoop latency threshold. That maximum latency is intended to be used by the processor to adjust various operational parameters so that the latency requirement can be met. Due to this erratum, the minimum latency value that can be specified via the Snoop Latency Multiplier field (bits[28:26]) and the Snoop Latency Value field (bits[25:16]) is 64 microseconds.

Implication: A minimum snoop latency requirement of 64 microseconds is so long that these registers are not useful.

Workaround: None identified. BIOS and the OS have other means to specify Package C-state exit latency maximums, which is the typical use model for setting PCIe* snoop latency limits.

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

BS144. A Machine Check May Result in an Unexpected Value in ECX

Problem: A machine check during execution of a REP MOVSB instruction may result in an unexpected value in ECX.

Implication: A machine check during execution of a REP MOVSB may result in an unexpected behavior.

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

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

BS145. System Hang May Occur when Memory Sparing is Enabled

Problem: Due to this erratum, enabling memory sparing can result in an internal timer error as indicated by the IA32_MCi_STATUS.MCACOD of 0000_0100_0000_0000.

Implication: Enabling memory sparing may result in a system may hang.

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

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

BS146. End Agent PCIe* Packet Errors May Result in a System Hang

Problem: PCIe agents are required by the PCIe Base Specification to identify and report packet errors. Due to this erratum, certain invalid completion types from the end agent are not correctly handled by the processor.

Implication: If a PCIe end agent issues certain invalid completion types, the system may hang.

Workaround: None identified

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

BS147. Retraining Cannot be Initiated by Downstream Devices in NTB/NTB or NTB/RP Configurations

Problem: The PCIe* Base Specification requires that a downstream device can initiate link retraining. Due to this erratum, link retraining cannot be initiated by the downstream device in a NTB/NTB (Non-Transparent Bridge) or a NTB/RP (Root Port) configuration.

Implication: The Retrain_Link field (LNKCON Device 3; Function 0; Offset 1A0H; bit [5]) does not function as expected in the identified configurations; software referencing the downstream device is not able to retrain the link.

Workaround: The link speed and training must be managed by the upstream host in NTB/NTB or NTB/RP configurations.

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

BS148. PCIe* Port in NTB Mode Flags Upstream Slot Power Limit Message as UR

Problem: When the processor is in NTB (Non-Transparent Bridge) mode, it should ignore upstream Slot Power Limit messages from the root port it is connected to. Due to this erratum, the processor generates UR (Unsupported Request) on these Slot Power Limit messages when in NTB mode.

Implication: Due to this erratum, some messages will be improperly flagged with UR.

Workaround: Upstream Slot Power Limit Message should be disabled in the identified configurations.

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

BS149. When in DMI Mode, Port 0's Device_Port_Type Field is Incorrect

Problem: When in DMI mode, the Device_Port_Type field (PXPCAP Bus 0; Device 0; Function 0; Offset 92H; bits [7:4]) should read as 9H (DMI mode) but incorrectly reads as 4H (PCIe mode).

Implication: Software may incorrectly conclude that this port is operating in PCIe mode when it is actually being used in the DMI mode.

Workaround: None identified.

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

BS150. PCIe* TPH Attributes May Result in Unpredictable System Behavior

Problem: TPH (Transactions Processing Hints) are optional aids to optimize internal processing of PCIe transactions. Due to this erratum, certain transactions with TPH attributes may be misdirected, resulting in unpredictable system behavior.

Implication: Use of the TPH feature may affect system stability.

Workaround: A BIOS workaround has been identified. Please refer to Intel® Romley Platform CPU/QPI/Memory Reference Code version 1.0.006 or later and release notes.

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

BS151. PCIe* Lane Reversal is Not Supported on All x8 Configurations During REUT Mode

Problem: PCIe lane reversal is not supported for Port 2 and Port 6 x8 configurations during REUT (Robust Electrical Unified Testing) mode.

Implication: Platforms that require REUT mode lane reversal for x8 Port 2 or Port 6 will not function per the PCIe Base Specification.

Workaround: None identified. Avoid designing platforms implementing lane reversal for x8 Port 2 and x8 Port 6 if REUT operation is needed.

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

BS152. PCIe* Port 3 Link Training May be Unreliable in NTB Mode

Problem: If PCIe port 3 is in NTB (Non-Transparent Bridge) mode and both the Root port and Endpoint Hardware Autonomous Speed Disable fields (LNKCON2 Bus 0; Device 3; Function 0; Offset 0C0H; bit 5) are set to 0, link training may fail. The Recovery.RcvrLock state may intermittently timeout and transition to the Detect state.

Implication: The NTB port link training may be unreliable.

Workaround: A BIOS workaround has been identified. Please refer to Intel® Romley Platform CPU/QPI/Memory Reference Code version 1.0.006 or later and release notes

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

BS153. A Machine-Check Exception Due to Instruction Fetch May Be Delivered Before an Instruction Breakpoint

Problem: Debug exceptions due to instruction breakpoints take priority over exceptions resulting from fetching an instruction. Due to this erratum, a machine-check exception resulting from the fetch of an instruction may take priority over an instruction breakpoint if the instruction crosses a 32-byte boundary and the second part of the instruction is in a 32-byte poisoned instruction fetch block.

Implication: Instruction breakpoints may not operate as expected in the presence of a poisoned instruction fetch block.

Workaround: None identified.

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

BS154. Intel® SpeedStep® Technology May Cause a System Hang

Problem: Intel SpeedStep Technology dynamically changes core operating frequencies. Due to this erratum, under complex conditions, core frequency changes may result in a system hang.

Implication: Intel SpeedStep Technology may cause a system hang.

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

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

BS155. The Accumulated Energy Status Read Service May Report a Power Spike Early in Boot

Problem: The PECI RdPkgConfig() command with an index value of 03H is the Accumulated Energy Status Read service. During platform boot, the Accumulated Energy Status Read service returns an accumulated energy value of 0. Later in the boot flow, due to this erratum, the Accumulated Energy Status Read service returns a value that is large. Energy values calculated with the first non-zero sample have been observed to be as high as 10kJ over a limited number of parts.

Implication: Software may interpret values returned by the Accumulated Energy Status Read service during boot time as indicating a large power spike. This could lead to unexpected or undesired platform power management actions.

Workaround: Once the first non-zero value is detected, the difference between subsequent sequential values is a reliable measure of energy consumed between the sample points.

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

BS156. Certain Uncorrectable Errors May Cause Loss of PECI Functionality

Problem: A PECI completion code of 91H indicates the PCU (Power Control Unit) detected an uncorrectable error that prevented processing of the PECI request. Due to this erratum, certain PCU or VRM error conditions may lead to a persistent 91H completion code for subsequent PECI request. Uncorrectable PCU errors are reported with an IA32_MC4_STATUS.MCACOD (MSR 411H, bits[15:0]) value of 0000_0100_0000_0010, IA32_MC4_STATUS.VALID (bit 63) set to 1, and IA32_MC4_STATUS.UC (bit 61) set to 1.

Implication: PECI processing may be blocked until either a cold reset or software running on one of the cores clears the IA32_MC4_STATUS register.

Workaround: None identified. Software running on one of the cores can clear the IA32_MC4_STATUS register to restore PECI functionality.

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

BS157. Machine Check During VM Exit May Result in VMX Abort

Problem: A machine check signaled during VM exit should cause a VMX abort only if the machine check would prevent successful completion of the VM exit; ordinarily, the machine

check should be delivered after the VM exit completes. Due to this erratum, certain machine checks (e.g., an uncorrectable cache error detected by another logical processor) may force a VM exit to result in a VMX abort even when that machine check does not interfere with the VM exit completing correctly.

Implication: Certain machine checks that could be reported in the host context for orderly logging and analysis may instead induce a VMX abort and shut down the logical processor.

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

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

BS158. Routing Intel® High Definition Audio Traffic Through VC1 May Result in System Hang

Problem: When bit 9 in the IIOMISCCTRL CSR (Bus 0; Device 5; Function 0; Offset 1C0H) is set, VCp inbound traffic (Intel® HD Audio) is routed through VC1 to optimize isochronous traffic performance. Due to this erratum, VC1 may not have sufficient bandwidth for all traffic routed through it; overflows may occur.

Implication: This erratum can result in lost completions that may cause a system hang.

Workaround: A BIOS workaround has been identified. Please refer to the latest version of the BIOS Spec Update, Intel® Romley Platform CPU/QPI/Memory Reference Code version 1.0.006 or later and release notes.

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

BS159. Package_Energy_Counter Register May Incorrectly Report Power Consumed by The Execution of Intel® AVX instructions

Problem: The processor includes a Package_Energy_Counter register to provide real-time energy consumption information. This facility can be accessed by the PECI RdPkgConfig() command with an index value of 03H (the Accumulated Energy Status Read service), by reading the PKG_ENERGY_STATUS MSR (611H) or by reading PACKAGE_ENERGY_STATUS CSR (Bus 1; Device 10; Function 0; Offset 90H). Due to this erratum, the power consumption reported during the execution of Intel AVX instructions is inaccurate.

Implication: Software that uses the Package_Energy_Counter register value during the execution of Intel AVX instructions may not behave as expected, possibly compromising thermal load balancing, processor throttling, or other platform management operations.

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

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

BS160. Coherent Interface Write Cache May Report False Correctable ECC Errors During Cold Reset

Problem: The Integrated I/O's coherent interface write cache includes ECC logic to detect errors. Due to this erratum, the write cache can report false ECC errors. This error is signaled by asserting bit 1 (Write Cache Corrected ECC) in the IRPP0ERRST CSR (Bus 0; Device 5; Function 2; Offset 230H) or the IRPP1ERRST CSR (Bus 0; Device 5; Function 2; Offset 2B0H).

Implication: If the coherent interface write cache ECC is enabled, the processor may incorrectly indicate correctable ECC errors in the write cache.

Workaround: A BIOS workaround has been identified. Please refer to Intel® Romley Platform CPU/QPI/Memory Reference Code version 1.0.006 or later and release notes

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

BS161. PCIe* RO May Result in a System Hang or Unpredictable System Behavior

Problem: PCIe RO (Relaxed Ordering) is not supported on this processor. Due to this erratum, enabling RO or, equivalently, not disabling RO throughout the Integrated I/O logic may lead to unpredictable system behavior or a system hang.

Implication: Enabling RO for any port or channel may lead to system instability.

Workaround: A BIOS workaround has been identified. Please refer to Intel® Romley Platform CPU/QPI/Memory Reference Code version 1.0.013 or later and release notes.

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

BS162. VT-d Invalidation Time-Out Error May Not be Signaled

Problem: Intel® VT-d (Virtualization Technology for Directed I/O) utilizes ITags to identify ATS (Address Translation Services) invalidation requests for invalidating Device-TLBs on endpoint devices. When an ATS invalidation response time-out is detected, the corresponding ITag is freed and an Invalidation Time-out Error is signaled through the VT-d Fault Status register. Due to this erratum, an ATS invalidation response timeout is detected and reported only for the first outstanding ITag entry.

Implication: As a result of the erratum, the ATS invalidation response timeout condition may not be reliably reported when multiple invalidation requests are outstanding. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

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

BS163. Enhanced Intel SpeedStep® Technology Hardware Coordination Cannot be Disabled

Problem: The processor should permit hardware coordination of Enhanced SpeedStep Technology requests to be disabled (then use the most recent P-state request from any core or logical processor to set the processor-wide P-state target). Due to this erratum, the Enhanced Intel SpeedStep Technology Hardware Coordination Disable value in bit 0 of the MISC_PWR_MGMT MSR (1AAH) is ignored; hardware coordination is always enabled.

Implication: It is not possible to prevent hardware P-state coordination.

Workaround: None identified.

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

BS164. PCIe* Link Upconfigure Capability is Incorrectly Advertised as Supported

Problem: The processor does not allow PCIe devices to dynamically change link width but, due to this erratum, the PCIe* Link Upconfigure Capability bit is incorrectly advertised as supported.

Implication: When a downstream device attempts to dynamically change the link's width, the link may not correctly retrain, resulting in an incorrect link width, reversed lane numbers, or Surprise Link Down (SLD).

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

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

BS165. The IA32_MCi_MISC.HaDbBank Field Should be Ignored

Problem: Home Agent parity errors, logged in IA32_MCi_STATUS.MCACOD (bits[15:0]) with a value of 0000_0000_1000_xxxx, may return an incorrect value in IA32_MCi_MISC.HaDbBank (bits[31:30]).

Implication: When analyzing Machine Check Register Bank contents, the IA32_MCi_MISC.HaDbBank field should be ignored.

Workaround: None identified.

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

BS166. When a PCIe* x4 Port Detects a Logical Lane 0 Failure, the Link Will Advertise Incorrect Lane Numbers

Problem: The PCIe interface incorporates a recovery mechanism for link degradation by retraining the link without affecting pending transactions. When a x4 port detects a lane failure on logical lane 0, the link degrades from x4 to x2 and lane reversal occurs. Due to this erratum, after degrading to x2 and reversing the lanes, the link will incorrectly advertise lane numbers as "PAD 0 1 0" instead of the correct "PAD PAD 1 0".

Implication: Devices that have the ability to negotiate a link with logical lane 0 on a mid physical lane may fail to successfully train the link.

Workaround: None identified.

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

BS167. Certain PCIe* TLPs May be Dropped

Problem: A PCIe TLP (Transaction Layer Packet) header can specify the request alignment (via byte enables), include TPH (Transaction Processing Hints), and request address translation via the AT field. Due to this erratum, a TLP with non-zero byte enables (i.e., not DWORD-aligned) that includes a non-zero TPH and with an AT field of "01" may be dropped.

Implication: Under the conditions noted, a PCIe TLP may be dropped, causing unpredictable system behavior. Intel has not observed this erratum with any commercially available software.

Workaround: Platforms must ensure that TPH and address translation requests are not used in the same TLP. The most direct means is to disable TPH in a PCIe device that may request an address translation. This can be accomplished by ensuring that TPH Requester Control Register (at offset 08H in the device's TPH Requester Capability structure) bits [9:8] are zero.

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

BS168. A Machine Check Exception Concurrent With an I/O SMI May Be Erroneously Reported as Restartable

Problem: A machine check exception that is delivered between the execution of an I/O instruction (IN, INS, OUT, or OUTS) and an SMI (system-management interrupt) triggered by that instruction may prevent proper handling of the SMI; because of this, the machine check exception should not be reported as restartable. Due to this erratum, such a machine check exception may be reported as restartable.

Implication: A restartable machine check exception on an I/O instruction concurrent with a resulting SMI may result in unpredictable system behavior.

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

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

BS169. VEX.L is Not Ignored with VCVT*2SI Instructions

Problem: The VEX.L bit should be ignored for the VCVTSS2SI, VCVTSD2SI, VCVTTSS2SI, and VCVTTSD2SI 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 1 with VCVTSS2SI, VCVTSD2SI, VCVTTSS2SI, and VCVTTSD2SI instructions.

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

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

BS170. The System Agent Temperature is Not Available

Problem: Due to this erratum, the processor does not record the temperature of the System Agent in the Temperature field in bits [7:0] of the SA_TEMPERATURE CSR (Device 10; Function 2; Offset: 044h).

Implication: Firmware cannot read the temperature of the System Agent via accessing the SA_TEMPERATURE CSR.

Workaround: None Identified. The System Agent temperature is available via PECI RdPkgConfig Command service, parameter value 00FFh.

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

BS171. The PCIe* Link at 8.0 GT/s is Transitioning Too Soon to Normal Operation While Training

Problem: The PCIe bus uses high speed serial links that must go through a training process to allow both transmitter and receiver to make adjustments in behavior to optimize the signaling between the transmitter and receiver. When a PCIe compliant device must train or retrain the link, training sequences are used. The device must allow enough time for the training to complete before transitioning to normal operation. In the case of PCIe equalization at 8.0 GT/s the processor is not allowing enough time to optimize signaling before attempting normal operation.

Implication: Due to this erratum, unexpected system behavior may be observed.

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

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

BS172. An ACM Error May Cause a System Power Down

Problem: An Intel® TXT (Trusted Executed Technology) enabled system that detects an ACM (Authenticated Code Module) error should perform a warm reset then start-up in non-trusted mode. Due to this erratum, an ACM error may cause the system to power down.

Implication: The system may unexpectedly power down.

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

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

BS173. Incorrect Retry Packets May Be Sent by a PCIe* x16 Port Operating at 8 GT/s

Problem: A PCIe x16 port operating at 8 GT/s transmitting 256 byte Completion TLPs may not replay TLPs correctly.

Implication: Due to this erratum, unpredictable system behavior may result when a 256 byte Completion TLP is replayed on a PCIe x16 port operating at 8 GT/s.

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

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

BS174. The Coherent Interface Error Codes "C2", "C3", "DA" and "DB" are Incorrectly Flagged

Problem: The Coherent Interface Error Status Registers (IRPP0ERRST and IRPP1ERRST at CPUBUS(0), Device 5, Function 2, Offsets 230H and 2B0H respectively) indicate that an error has been detected by the Coherent Interface.Bit 3 indicates that a Write Cache Un-correctable ECC (C2) error has occurred.Bit 4 indicates that a CSR access crossing 32-bit boundary (C3) error has occurred.Bit 13 indicates that a Protocol Queue/Table Overflow or Underflow (DA) error has occurred.Bit 14 indicates that a Protocol Parity

Error (DB) error has occurred.Due to this erratum, the processor may incorrectly log the "C2", "C3", "DA" and "DB" error flags.

Implication: The "C2", "C3", "DA" and "DB" error flags are indeterminate.

Workaround: Mask off the "C2", "C3", "DA" and "DB" error flags (bit 3, bit 4, bit 13 and bit 14) of the IRPP0ERRCTL and IRPP1ERRCTL registers at CPUBUS(0), Device 5, Function 2, Offsets 234H and 2B4H respectively

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

BS175. MCI_ADDR May be Incorrect For Cache Parity Errors

Problem: In cases when a WBINVD instruction evicts a line containing an address or data parity error (MCACOD of 0x124, and MSCOD of 0x10), the address of this error should be logged in the MCi_ADDR register. Due to this erratum, the logged address may be incorrect, even though MCi_Status.ADDRV (bit 63) is set.

Implication: The address reported in MCi_ADDR may not be correct for cases of a parity error found during WBINVD execution.

Workaround: None identified.

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

BS176. Intel® QuickData DMA Channel Write Abort Errors May Cause a Channel Hang

Problem: When the "Fence" bit in the base descriptor Control field is set, the DMA engine assures all data for that operation (and previous operations) has been written before considering a transfer complete and beginning to process the next chained base descriptor. In addition, upon completion of a transfer, the DMA engine can notify software of the completion via either an interrupt, a memory write to a programmed location, or both. Due to this erratum, the DMA engine, while processing chained DMA descriptors with fencing or interrupt completion enabled, may hang and not enter the HALT state as expected if a write error that results in an abort occurs.

Implication: A DMA transfer that suffers a write abort error when fencing or interrupt completion is enabled may hang.

Workaround: Do not enable fencing bit [4] or interrupt completion bit [0] in the Descriptor Control Field.

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

BS177. The Processor May Not Properly Execute Code Modified Using A Floating-Point Store

Problem: Under complex internal conditions, a floating-point store used to modify the next sequential instruction may result in the old instruction being executed instead of the new instruction.

Implication: Self- or cross-modifying code may not execute as expected. Intel has not observed this erratum with any commercially available software.

Workaround: None identified. Do not use floating-point stores to modify code.

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

BS178. Execution of GETSEC[SEXIT] May Cause a Debug Exception to be Lost

Problem: A debug exception occuring at the same time that GETSEC[SEXIT] is executed or when an SEXIT doorbell event is serviced may be lost.

Implication: Due to this erratum, there may be a loss of a debug exception when it happens concurrently with the execution of GETSEC[SEXIT]. Intel has not observed this erratum with any commercially available software.

Workaround: None identified.

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

BS179. VM Exits Due to GETSEC May Save an Incorrect Value for "Blocking by STI" in the Context of Probe-Mode Redirection

Problem: The GETSEC instruction causes a VM exit when executed in VMX non-root operation. Such a VM exit should set bit 0 in the interruptibility-state field in the virtual-machine control structure (VMCS) if the STI instruction was blocking interrupts at the time GETSEC commenced execution. Due to this erratum, a VM exit executed in VMX non-root operation may erroneously clear bit 0 if redirection to probe mode occurs on the GETSEC instruction.

Implication: After returning from probe mode, a virtual interrupt may be incorrectly delivered prior to GETSEC instruction. Intel has not observed this erratum with any commercially software.

Workaround: None identified.

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

BS180. Warm Reset May Cause PCIe* Hot-Plug to Fail

Problem: The Integrated I/O unit uses the VPP (Virtual Pin Port) to communicate with devices that interface to the switches and LEDs associated with PCIe Hot-Plug sequencing. Due to this erratum, VPP operation stalls when a warm reset occurs and then experiences delayed reset. Depending on timing alignment with the warm reset event, a VPP transaction in progress around the time of a warm reset may suffer an extended stall or an immediate termination.

Implication: Hot-Plug sequencing may suffer failures during or shortly after warm resets which may be temporary or persist until the next cold reset.

Workaround: None identified.

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

BS181. Certain Local Memory Read / Load Retired PerfMon Events May Undercount

Problem: Due to this erratum, the Local Memory Read / Load Retired PerfMon events listed below may undercount.

MEM_LOAD_UOPS_RETIRED.LLC_HIT
MEM_LOAD_UOPS_RETIRED.LLC_MISS*
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_MISS
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HIT
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HITM
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_NONE
MEM_LOAD_UOPS_LLC_MISS_RETIRED.LOCAL_DRAM*
MEM_LOAD_UOPS_LLC_MISS_RETIRED.REMOTE_DRAM*
MEM_TRANS_RETIRED.LOAD_LATENCY*

Implication: The affected events may undercount, resulting in inaccurate memory profiles. The undercount of these events can be partially resolved (but not eliminated) by setting MSR_PEBS_NUM_ALT. PEBS Accuracy Enable (MSR 39CH; bit 0) to 1. When using the events marked with an asterisk, set the Direct-to-core disable field (Bus 1; Device 14; Function 0; Offset 84; bit 1) to 1 for Local memory reads and (Bus 1; Device 8; Function 0; Offset 80; bit 1) to 1 and (Bus 1; Device 9; Function 0; Offset 80; bit 1) to 1 for Remote memory reads. The improved accuracy comes at the cost of a reduction in performance; this workaround generally should not be used during normal operation.

Workaround: None identified.

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

BS182. Performance Monitor Counters May Produce Incorrect Results

Problem: When operating in hyper-threaded mode, a memory at-retirement performance monitoring event (from the list below) may be dropped or may increment an enabled counter on the physical core's other thread rather than the thread experiencing the event.

The list of affected memory at-retirement events is as follows:

MEM_UOP_RETIRED.LOADS
MEM_UOP_RETIRED.STORES
MEM_UOP_RETIRED.LOCK
MEM_UOP_RETIRED.SPLIT
MEM_UOP_RETIRED.STLB_MISS
MEM_LOAD_UOPS_RETIRED.HIT_LFB
MEM_LOAD_UOPS_RETIRED.L1_HIT
MEM_LOAD_UOPS_RETIRED.L2_HIT
MEM_LOAD_UOPS_RETIRED.LLC_HIT
MEM_LOAD_UOPS_MISC_RETIRED.LLC_MISS
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HIT
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_HITM
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_MISS
MEM_LOAD_UOPS_LLC_HIT_RETIRED.XSNP_NONE
MEM_LOAD_UOPS_RETIRED.LLC_MISS
MEM_LOAD_UOPS_LLC_MISS_RETIRED.LOCAL_DRAM
MEM_LOAD_UOPS_LLC_MISS_RETIRED.REMOTE_DRAM
MEM_LOAD_UOPS_RETIRED.L2_MISS

Implication: Due to this erratum, certain performance monitoring event will produce unreliable results during hyper-threaded operation.

Workaround: None identified.

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

BS183. The Corrected Error Count Overflow Bit in IA32_ MC0_STATUS is Not Updated After a UC Error is Logged

Problem: When a UC (uncorrected) error is logged in the IA32_MC0_STATUS MSR (401H), corrected errors will continue to update 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 after a UC error is logged.

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, see the Summary Tables of Changes.

BS184. Spurious Intel® VT-d Interrupts May Occur When the PFO Bit is Set

Problem: When the PFO (Primary Fault Overflow) field (bit [0] in the VT-d FSTS [Fault Status] register) is set to 1, further faults should not generate an interrupt. Due to this erratum, further interrupts may still occur.

Implication: Unexpected Invalidation Queue Error interrupts may occur. Intel has not observed this erratum with any commercially available software.

Workaround: Software should be written to handle spurious Intel® VT-d fault interrupts.

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

BS185. 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 Summary Tables of Changes.

Specification Changes

There are no Specification Changes at this time.

Specification Clarifications

There are no Specification Clarifications at this time.

Documentation Changes

BS1. On-Demand Clock Modulation Feature Clarification

Software Controlled Clock Modulation section of the Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3B: System Programming Guide will be modified to differentiate On-demand clock modulation feature on different processors. The clarification will state:

For Hyper-Threading Technology enabled processors, the IA32_CLOCK_MODULATION register is duplicated for each logical processor. In order for the On-demand clock modulation feature to work properly, the feature must be enabled on all the logical processors within a physical processor. If the programmed duty cycle is not identical for all the logical processors, the processor clock will modulate to the highest duty cycle programmed for processors if the CPUID DisplayFamily_DisplayModel signatures is listed in Table 14-2. For all other processors, if the programmed duty cycle is not identical for all logical processors in the same core, the processor will modulate at the lowest programmed duty cycle.

For multiple processor cores in a physical package, each core can modulate to a programmed duty cycle independently.

For the P6 family processors, on-demand clock modulation was implemented through the chipset, which controlled clock modulation through the processor's STPCLK# pin.

Table 14-2. CPUID Signatures for Legacy Processors That Resolve to Higher Performance Setting of Conflicting Duty Cycle Requests

DisplayFamily_Display Model	DisplayFamily_Display Model	DisplayFamily_Display Model	DisplayFamily_Display Model
0F_xx	06_1C	06_1A	06_1E
06_1F	06_25	06_26	06_27
06_2C	06_2E	06_2F	06_35
06_36			

§ §