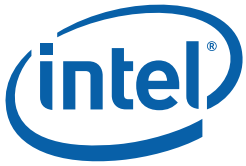


Intel® Pentium® D Processor 900Δ Sequence and Intel® Pentium® Processor Extreme Edition 955, 965Δ

Specification Update

- On 65 nm Process in the 775-land LGA Package supporting Intel® 64 Architecture and Intel® Virtualization Technology±

May 2008

Document Number: 310307-018

Contents

Revision History

Version	Description	Date
-001	• Initial release	December 2005
-002	• Added Intel Pentium D processor 900 sequence specifications	January 2006
-003	• Updated related documents, added processor number	February 2006
-004	• Replaced AA5 with new erratum, added erratum AA31	March 2006
-005	• Added Intel® Pentium® Processor Extreme Edition 965	Out of Cycle March 22, 2006
-006	• Added Errata AA32, AA33, AA34, added C1 step	April 2006
-007	• Added Pentium D processor 960	Out of Cycle May 1, 2006
-008	• Added Erratum AA35	May 2006
-009	• Added Errata AA36-41	June 2006
-010	• Added Rtt Specification change	July 2006
-011	• Added Intel Pentium D processor 945 and 915	Out of Cycle July 26, 2006
-012	• Added Erratum AA42	September 2006
-013	• Added Intel Pentium D processor 925	September 22 2006
-014	• Added D0-step	December 2006
-015	• Added Intel Pentium D processor 935	January 2007
-016	• Updated names of Software Developers Manuals • Updated Summary table of changes • Added Erratum AA43	April 2007
-017	• Added AA44	July 2007
-018	• Added AA45	May 2008

§

Preface

This document is an update to the specifications contained in the documents listed in the following Affected Documents/Related Documents table. It is a compilation of device and document errata and specification clarifications and changes, and is intended for hardware system manufacturers and for software developers of applications, operating system, and tools.

Information types defined in the Nomenclature section of this document are consolidated into this update document and are no longer published in other documents. This document may also contain information that has not been previously published.

Affected Documents

Document Title	Document Number
Intel® Pentium® D Processor 900$^{\Delta}$ Sequence and Intel® Pentium® Processor Extreme Edition 955$^{\Delta}$, 965$^{\Delta}$ Datasheet	310306-003

Related Documents

Document Title	Document Number
Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 1: Basic Architecture, document 253665	http://www.intel.com/products/processor/manuals/index.htm
Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 2A: Instruction Set Reference Manual A–M, document 253666	http://www.intel.com/products/processor/manuals/index.htm
Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 2B: Instruction Set Reference Manual, N–Z, document 253667	http://www.intel.com/products/processor/manuals/index.htm
Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3A: System Programming Guide, document 253668	http://www.intel.com/products/processor/manuals/index.htm
Intel® 64 and IA-32 Architectures Software Developer's Manual Volume 3B: System Programming Guide, document 253669	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 (e.g., core speed, L2 cache size, package type, etc.) as described in the processor identification information table. Care should be taken to read all notes associated with each S-Spec number

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

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

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

Specification Clarifications describe a specification in greater detail or further highlight a specification's impact to a complex design situation. These clarifications will be incorporated in the next release of the specifications.

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

Note: Errata remain in the specification update throughout the product's lifecycle, or until a particular stepping is no longer commercially available. Under these circumstances, errata removed from the specification update are archived and available upon request. Specification changes, specification clarifications and documentation changes are removed from the specification update when the appropriate changes are made to the appropriate product specification or user documentation (datasheets, manuals, etc.).

§

Summary Tables of Changes

The following table indicates the Specification Changes, Errata, Specification Clarifications or Documentation Changes, which apply to the listed MCH steppings. Intel intends to fix some of the errata in a future stepping of the component, and to account for the other outstanding issues through documentation or Specification Changes as noted. This table uses the following notations:

Codes Used in Summary Table

Stepping

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

Status

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

Row

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

Item Numbering

Each Specification Update item is prefixed with a capital letter to distinguish the product. The key below details the letters that are used in Intel's microprocessor specification updates:

A = Dual-Core Intel® Xeon® processor 7000 sequence
C = Intel® Celeron® processor
D = Dual-Core Intel® Xeon® processor 2.80 GHz
E = Intel® Pentium® III processor
F = Intel® Pentium® processor Extreme Edition and Intel® Pentium® D processor
I = Dual-Core Intel® Xeon® processor 5000 series
J = 64-bit Intel® Xeon® processor MP with 1MB L2 cache
K = Mobile Intel® Pentium® III processor
L = Intel® Celeron® D processor
M = Mobile Intel® Celeron® processor
N = Intel® Pentium® 4 processor
O = Intel® Xeon® processor MP
P = Intel ® Xeon® processor
Q = Mobile Intel® Pentium® 4 processor supporting Hyper-Threading technology on 90-nm process technology
R = Intel® Pentium® 4 processor on 90 nm process
S = 64-bit Intel® Xeon® processor with 800 MHz system bus (1 MB and 2 MB L2 cache versions)
T = Mobile Intel® Pentium® 4 processor-M
U = 64-bit Intel® Xeon® processor MP with up to 8MB L3 cache
V = Mobile Intel® Celeron® processor on .13 micron process in Micro-FCPGA package
W= Intel® Celeron® M processor
X = Intel® Pentium® M processor on 90nm process with 2-MB L2 cache and Intel® processor A100 and A110 with 512-KB L2 cache
Y = Intel® Pentium® M processor
Z = Mobile Intel® Pentium® 4 processor with 533 MHz system bus
AA = Intel® Pentium® D processor 900 sequence and Intel® Pentium® processor Extreme Edition 955, 965
AB = Intel® Pentium® 4 processor 6x1 sequence
AC = Intel(R) Celeron(R) processor in 478 pin package
AD = Intel(R) Celeron(R) D processor on 65nm process
AE = Intel® Core™ Duo processor and Intel® Core™ Solo processor on 65nm process
AF = Dual-Core Intel® Xeon® processor LV
AG = Dual-Core Intel® Xeon® processor 5100 series
AH = Intel® Core™2 Duo/Solo processor for Intel® Centrino® Duo processor technology
AI = Intel® Core™2 Extreme processor X6800 and Intel® Core™2 Duo desktop processor E6000 and E4000 sequence
AJ = Quad-Core Intel® Xeon® processor 5300 series

AK =	Intel® Core™2 Extreme quad-core processor QX6000 sequence and Intel® Core™2 Quad processor Q6000 sequence
AL =	Dual-Core Intel® Xeon® processor 7100 series
AM =	Intel® Celeron® processor 400 sequence
AN =	Intel® Pentium® dual-core processor
AO =	Quad-Core Intel® Xeon® processor 3200 series
AP =	Dual-Core Intel® Xeon® processor 3000 series
AQ =	Intel® Pentium® dual-core desktop processor E2000 sequence
AR =	Intel® Celeron® processor 500 series
AS =	Intel® Xeon® processor 7200, 7300 series
AV =	Intel® Core™2 Extreme processor QX9650 and Intel® Core™2 Quad processor Q9000 series
AW =	Intel® Core™ 2 Duo processor E8000 series
AX =	Quad-Core Intel® Xeon® processor 5400 series
AY=	Dual-Core Intel® Xeon® processor 5200 series
AZ =	Intel® Core™2 Duo Processor and Intel® Core™2 Extreme Processor on 45-nm Process
AAA =	Quad-Core Intel® Xeon® processor 3300 series
AAB =	Dual-Core Intel® Xeon® E3110 Processor
AAC =	Intel® Celeron® dual-core processor E1000 series
AAD =	Intel® Core™2 Extreme Processor QX9775Δ
AAE =	Intel® Atom™ processor Z5xx series

NO	B1	C1	D0	Plan	ERRATA
AA1	X	X	X	No Fix	Locks and SMC Detection May Cause the Processor to Temporarily Hang
AA2	X	X	X	No Fix	Memory Aliasing of Pages as Uncacheable Memory Type and Write Back (WB) May Hang the System
AA3	X	X	X	No Fix	Data Breakpoints on the High Half of a Floating Point Line Split may not be Captured
AA4	X	X	X	No Fix	MOV CR3 Performs Incorrect Reserved Bit Checking When in PAE Paging
AA5	X	X	X	Plan Fix	VMEntry from 64-bit Host to 32-bit Guest may Cause IERR# with Hyper-Threading Enabled
AA6	X	X	X	No Fix	FXRSTOR May Not Restore Non-canonical Effective Addresses on Processors with Intel® Extended Memory 64 Technology (Intel® EM64T) Enabled
AA7	X	X	X	No Fix	A Push of ESP that Faults may Zero the Upper 32 Bits of RSP
AA8	X	X	X	No Fix	Checking of Page Table Base Address May Not Match the Address Bit Width Supported by the Platform
AA9	X	X	X	No Fix	With TF (Trap Flag) Asserted, FP Instruction That Triggers an Unmasked FP Exception May Take Single Step Trap Before Retirement of Instruction

NO	B1	C1	D0	Plan	ERRATA
AA10	X	X	X	No Fix	BTS (Branch Trace Store) and PEBS(Precise Event Based Sampling) May Update Memory outside the BTS/PEBS Buffer
AA11	X	X	X	No Fix	Control Register 2 (CR2) Can be Updated during a REP MOVS/STOS Instruction with Fast Strings Enabled
AA12	X	X	X	No Fix	REP STOS/MOVS Instructions with RCX >=2^32 May Cause a System Hang
AA13	X	X	X	No Fix	A 64-Bit Value of Linear Instruction Pointer (LIP) May be Reported Incorrectly in the Branch Trace Store (BTS) Memory Record or in the Precise Event Based Sampling (PEBS) Memory Record
AA14	X	X	X	No Fix	Access to an Unsupported Address Range in Uniprocessor (UP) or Dual-processor (DP) Systems Supporting Intel® Virtualization Technology May Not Trigger Appropriate Actions
AA15	X	X		Fixed	VM Exit Due to a MOV from CR8 May Cause an Unexpected Memory Access
AA16	X	X	X	Plan Fix	The Processor May Incorrectly Respond to Machine Checks during VM Entry/Exit Transitions
AA17	X	X		Fixed	Power Down Requests May not be Serviced if a Power Down Transition is Interrupted by an In-Target Probe Event in the Presence of a Specific Type of VM Exit
AA18	X	X		Fixed	VM EXIT Due to TPR shadow Below Threshold May Improperly Set and Cause "Blocking by STI" actions
AA19	X	X	X	No Fix	Two Correctable L2 Cache Errors in Close Proximity May Cause a System Hang
AA20	X	X		Fixed	A VM Exit due to SMI or INIT in Parallel with a Pending FP Exception May Not Correctly Clear the Interruptibility State Bits
AA21	X	X	X	No Fix	Processor May Hang with a 25% or Less STPCLK# Duty Cycle
AA22	X	X		Fixed	Attempting to Use an LDT Entry when the LDTR Has Been Loaded with an Unusable Segment May Cause Unexpected Memory Accesses
AA23	X	X	X	No Fix	Machine Check Exceptions May not Update Last-Exception Record MSRs (LERs)
AA24	X	X		Fixed	VM Entry/Exit Writes to LSTAR/SYSCALL_FLAG MSR's May Cause Incorrect Data to be Written to Bits [63:32]
AA25	X	X	X	No Fix	Writing the Local Vector Table (LVT) when an Interrupt is Pending May Cause an Unexpected Interrupt
AA26	X	X		Fixed	At a Bus Ratio of 13:1, RCNT and Address Parity May be Incorrect
AA27	X	X		Fixed	The Execution of a VMPTRLD Instruction May Cause an Unexpected Memory Access
AA28	X	X		Fixed	The Execution of VMPTRLD or VMREAD May Cause an Unexpected Memory Access
AA29	X	X		Fixed	On a "Failed VM-entry" VM Exit, the VMCS Pointer May have Incorrect Value

NO	B1	C1	D0	Plan	ERRATA
AA30	X			Fixed	During an Enhanced HALT or Enhanced Intel® Speed Step Technology Ratio Transition the System May Hang
AA31	X	X	X	No Fix	L2 Cache ECC Machine Check Errors May be erroneously Reported after an Asynchronous RESET# Assertion
AA32	X	X	X	Plan Fix	VMCALL to Activate Dual-monitor Treatment of SMIs and SMM Ignores Reserved Bit settings in VM-exit Control Field
AA33	X	X	X	No Fix	Using 2M/4M Pages When A20M# Is Asserted May Result in Incorrect Address Translations
AA34	X	X	X	No Fix	Writing Shared Unaligned Data that Crosses a Cache Line without Proper Semaphores or Barriers May Expose a Memory Ordering Issue
AA35	X	X	X	No Fix	The IA32_MC0_STATUS and IA32_MC1_STATUS Overflow Bit is not set when Multiple Un-correctable Machine Check Errors Occur at the Same Time
AA36	X	X	X	No Fix	IRET under Certain Conditions May Cause an Unexpected Alignment Check Exception
AA37	X	X	X	No Fix	Processor May Fault When the Upper 8 Bytes of Segment Selector Is Loaded from a Far Jump through a Call Gate via the Local Descriptor Table
AA38	X	X	X	No Fix	The Processor May Issue Front Side Bus Transactions up to 6 Clocks after RESET# is Asserted
AA39	X	X	X	No Fix	Front Side Bus Machine Checks May be Reported as a Result of On-Going Transactions during Warm Reset
AA40	X	X	X	No Fix	NMI-blocking Information Recorded in VMCS May be Incorrect after a #GP on an IRET Instruction
AA41	X	X	X	No Fix	VMLAUNCH/VMRESUME May Not Fail when VMCS is Programmed to Cause VM Exit to Return to a Different Mode
AA42	X	X	X	No Fix	A Continuous Loop Executing Bus Lock Transactions on One Logical Processor may Prevent Another Logical Processor from Acquiring Resources
AA43			X	No Fix	An Unexpected Memory Access May be Issued During Execution of the WRMSR Instruction Under Certain Conditions
AA44			X	Plan Fix	Combining Some Processors With Intel 945® Chipsets Can Lead to Unpredictable System Behavior
AA45	X	X	X	No Fix	A VM Exit Occuring in IA-32e Mode May Not Produce a VMX Abort When Expected

Number	Plan	SPECIFICATION CHANGES
		There are no Specification Changes in this Specification Update revision

Number	Plan	SPECIFICATION CLARIFICATIONS
		There are no Specification Clarification in this Specification Update revision

Number	Plan	DOCUMENTATION CHANGES
		There are no Documentation Changes in this Specification Update revision.

§

General Information

Figure 1. Intel® Pentium® D Processor 900 Sequence (Package Top Markings)

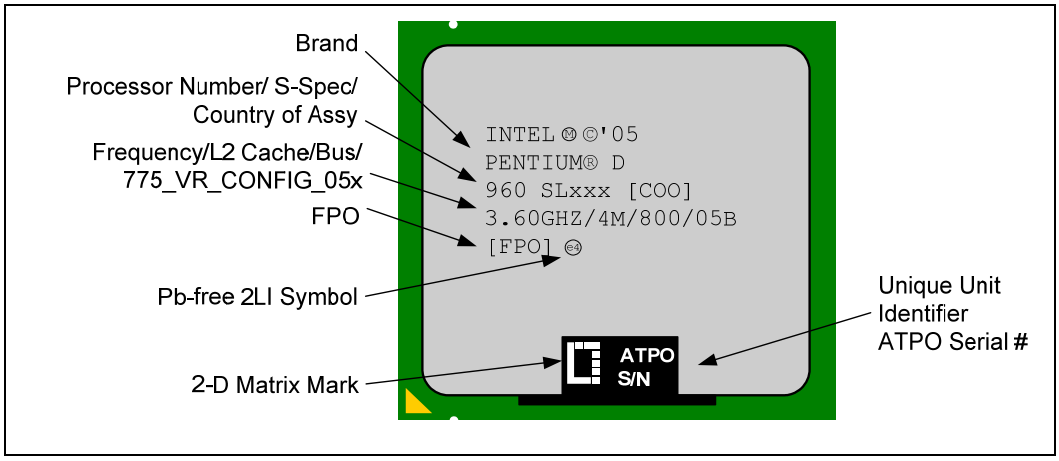
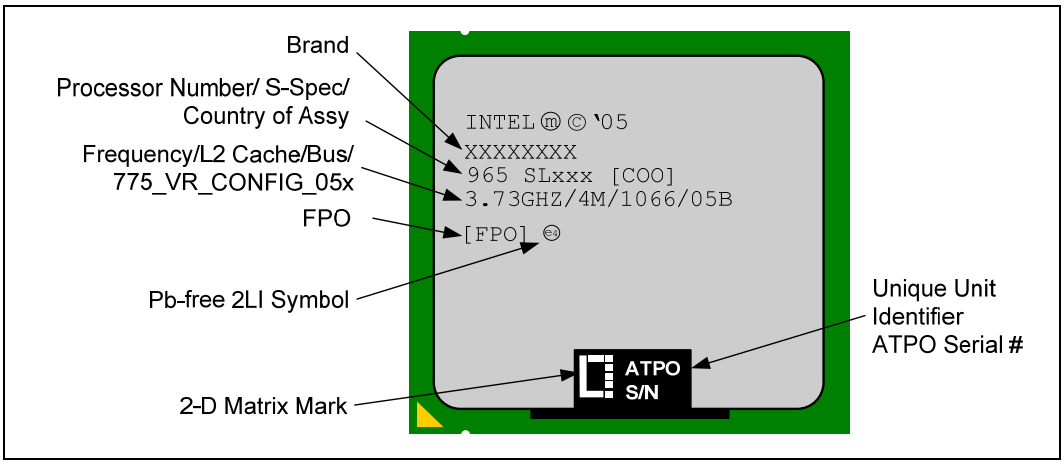


Figure 2. Intel® Pentium® Processor Extreme Edition 965 (Package Top Markings)



§

Identification Information

The Intel® Pentium® D processor 900 sequence and Intel® Pentium® processor Extreme Edition 955, 965 and can be identified by the following values:

Family[1]	Model[2]
1111b	0110b

NOTES:
1. The Family 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.
2. The Model 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.

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. Refer to the *Intel Processor Identification and the CPUID Instruction Application Note* (AP-485).

Table 1. Intel® Pentium® D Processor 900 Sequence and Intel® Pentium® Processor Extreme Edition 955, 965 Identification Information

S-Spec	Core Stepping	L2 Cache Size (bytes)	Processor Signature	Processor Number	Speed Core / Bus	Package	Notes
SL8WS	B1	2M x 2	0F62h	920	2.80GHz/800MHz	775-land LGA	1, 3, 6, 7
SL8WR	B1	2M x 2	0F62h	930	3.00GHz/800MHz	775-land LGA	1, 3, 6, 7
SL8WQ	B1	2M x 2	0F62h	940	3.20GHz/800MHz	775-land LGA	2, 3, 6, 7
SL8WP	B1	2M x 2	0F62h	950	3.40GHz/800MHz	775-land LGA	2, 3, 6, 7
SL8WM	B1	2M x 2	0F62h	Extreme Edition 955	3.46GHz/1066MHz	775-land LGA	2, 4, 6, 7
SL94S	B1	2M x 2	0F62h	920	2.80GHz/800MHz	775-land LGA	1, 5, 6
SL94R	B1	2M x 2	0F62h	930	3.00GHz/800MHz	775-land LGA	1, 5, 6
SL94Q	B1	2M x 2	0F62h	940	3.20GHz/800MHz	775-land LGA	2, 5, 6
SL94P	B1	2M x 2	0F62h	950	3.40GHz/800MHz	775-land LGA	2, 5, 6
SL94N	B1	2M x 2	0F62h	Extreme Edition 955	3.46GHz/1066MHz	775-land LGA	2, 4, 5, 6
SL9DA	C1	2M x 2	0F64h	915	2.80GHz/800MHz	775-Land LGA	1, 3

Table 1. Intel® Pentium® D Processor 900 Sequence and Intel® Pentium® Processor Extreme Edition 955, 965 Identification Information

S-Spec	Core Stepping	L2 Cache Size (bytes)	Processor Signature	Processor Number	Speed Core / Bus	Package	Notes
SL9D9	C1	2M x 2	0F64h	925	3.00GHz/800MHz	775-Land LGA	1, 3
SL9QB	C1	2M x 2	0F64h	945	3.40GHz/800MHz	775-Land LGA	1, 3
SL95X	C1	2M x 2	0F64h	930	3.00GHz/800MHz	775-Land LGA	1, 3, 6
SL95W	C1	2M x 2	0F64h	940	3.20GHz/800MHz	775-Land LGA	1, 3, 6
SL95V	C1	2M x 2	0F64h	950	3.40GHz/800MHz	775-Land LGA	1, 3, 6
SL9AP	C1	2M x 2	0F64h	960	3.60GHz/800MHz	775-Land LGA	2, 3, 6
SL9AN	C1	2M x 2	0F64h	Extreme Edition 965	3.73GHz/1066MHz	775-land LGA	2, 4, 6
SL9KB	D0	2M x 2	0F65h	915	2.80GHz/800MHz	775-Land LGA	1, 3
SL9KA	D0	2M x 2	0F65h	925	3.00GHz/800MHz	775-Land LGA	1, 3
SL9QR	D0	2M x 2	0F65h	935	3.20GHz/800MHz	775-Land LGA	1, 3
SL9QQ	D0	2M x 2	0F65h	945	3.40GHz/800MHz	775-Land LGA	1, 3
SL9K8	D0	2M x 2	0F65h	950	3.40GHz/800MHz	775-Land LGA	1, 3, 6
SL9K7	D0	2M x 2	0F65h	960	3.60GHz/800MHz	775-Land LGA	1, 3, 6

NOTES:

1. These processors support the 775_VR_CONFIG_05A (mainstream) specifications.
2. These processors support the 775_VR_CONFIG_05B (performance) specifications.
3. These parts support Enhanced Intel SpeedStep® Technology and Enhanced HALT State.
4. These parts support Hyper-Threading Technology.
5. These parts do NOT support Enhanced Intel SpeedStep® Technology or Enhanced HALT State
6. These parts support Intel® Virtualization Technology
7. These parts are affected by erratum AA30

§

Errata

AA1. Bus Locks and SMC Detection May Cause the Processor to Hang Temporarily

Problem: The processor may temporarily hang in an Hyper-Threading Technology enabled system if one logical processor executes a synchronization loop that includes one or more locks and is waiting for release by the other logical processor. If the releasing logical processor is executing instructions that are within the detection range of the self -modifying code (SMC) logic, then the processor may be locked in the synchronization loop until the arrival of an interrupt or other event.

Implication: If this erratum occurs in an HT Technology enabled system, the application may temporarily stop making forward progress. 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*.

AA2. Memory Aliasing of Pages As Uncacheable Memory Type and Write Back (WB) May Hang the System

Problem: When a page is being accessed as either Uncacheable (UC) or Write Combining (WC) and WB, under certain bus and memory timing conditions, the system may loop in a continual sequence of UC fetch, implicit writeback, and Request For Ownership (RFO) retries.

Implication: This erratum has not been observed in any commercially available operating system or application. The aliasing of memory regions, a condition necessary for this erratum to occur, is documented as being unsupported in the *IA-32 Intel® Architecture Software Developer's Manual*, Volume 3, section 10.12.4, Programming the PAT. However, if this erratum occurs the system may hang.

Workaround: The pages should not be mapped as either UC or WC and WB at the same time.

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

AA3. Data Breakpoints on the High Half of a Floating Point Line Split May Not Be Captured

Problem: When a floating point load which splits a 64-byte cache line gets a floating point stack fault, and a data breakpoint register maps to the high line of the floating point load, internal boundary conditions exist that may prevent the data breakpoint from being captured.

Implication: When this erratum occurs, a data breakpoint will not be captured.

Workaround: None identified.

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

AA4. MOV CR3 Performs Incorrect Reserved Bit Checking When in PAE Paging

Problem: The MOV CR3 instruction should perform reserved bit checking on the upper unimplemented address bits. This checking range should match the address width reported by CPUID instruction 0x8000008. This erratum applies whenever PAE is enabled.

Implication: Software that sets the upper address bits on a MOV CR3 instruction and expects a fault may fail. This erratum has not been observed with commercially available software.

Workaround: None identified.

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

AA5. VMEntry from 64-bit Host to 32-bit Guest may Cause IERR# with Hyper-Threading Technology Enabled

Problem: When transitioning from a 64-bit host environment to a 32-bit guest environment via a VMEntry, internal conditions in a processor with Hyper-Threading enabled may cause a speculative page-table walk to be prematurely terminated, resulting in a processor hang and the assertion of IERR#.

Implication: An IERR# may occur on VMEntry from a 64-bit to a 32-bit environment with Hyper-Threading enabled.

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.*

AA6. FXRSTOR May Not Restore Non-canonical Effective Addresses on Processors with Intel® Extended Memory 64 Technology (Intel® EM64T) Enabled

Problem: If an x87 data instruction has been executed with a non-canonical effective address, FXSAVE may store that non-canonical FP Data Pointer (FDP) value into the save image. An FXRSTOR instruction executed with 64-bit operand size may signal a General Protection Fault (#GP) if the FDP or FP Instruction Pointer (FIP) is in non-canonical form.

Implication: When this erratum occurs, Intel EM64T enabled systems may encounter an unintended #GP fault.

Workaround: Software should avoid using non-canonical effective addressing in EM64T enabled processors. BIOS can contain a workaround for this erratum removing the unintended #GP fault on FXRSTOR.

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

AA7. A Push of ESP That Faults May Zero the Upper 32 Bits of RSP

Problem: In the event that a push ESP instruction, that faults, is executed in compatibility mode, the processor will incorrectly zero upper 32-bits of RSP.

Implication: A Push of ESP in compatibility mode will zero the upper 32-bits of RSP. Due to this erratum, this instruction fault may change the contents of RSP. This erratum has not been observed in commercially available software.

Workaround: None identified.

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

AA8. Checking of Page Table Base Address May Not Match the Address Bit Width Supported by the Platform

Problem: If the page table base address, included in the page map level-4 table, page-directory pointer table, page-directory table or page table, exceeds the physical address range supported by the platform (e.g. 36-bit) and it is less than the implemented address range (e.g. 40-bit), the processor does not check if the address is invalid.

Implication: If software sets such invalid physical address in those tables, the processor does not generate a page fault (#PF) upon access to that virtual address, and the access results in an incorrect read or write. If BIOS provides only valid physical address ranges to the operating system, this erratum will not occur.

Workaround: BIOS must provide valid physical address ranges to the operating system.

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

AA9. With TF (Trap Flag) Asserted, FP Instruction That Triggers an Unmasked FP Exception May Take Single Step Trap before Retirement of Instruction

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

Implication: A Single Step trap will be taken when not expected.

Workaround: None identified.

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

AA10. BTS (Branch Trace Store) and PEBS (Precise Event Based Sampling) May Update Memory outside the BTS/PEBS Buffer

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

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

A BTS/PEBS record can be written that will wrap at the 4G boundary (IA32) or 2^64 boundary (EM64T mode), and write memory outside of the BTS/PEBS buffer.

Implication: Software that uses BTS/PEBS near the 4G boundary (IA32) or 2^64 boundary (EM64T mode), and defines the buffer such that it does not hold an integer multiple of records can update memory outside the BTS/PEBS buffer.

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

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

AA11. Control Register 2 (CR2) Can be Updated during a REP MOVS/STOS Instruction with Fast Strings Enabled

Problem: Under limited circumstances while executing a REP MOVS/STOS string instruction, with fast strings enabled, it is possible for the value in CR2 to be changed as a result of an interim paging event, normally invisible to the user. Any higher priority architectural event that arrives and is handled while the interim paging event is occurring may see the modified value of CR2.

Implication: The value in CR2 is correct at the time that an architectural page fault is signaled. 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*.

AA12. REP STOS/MOVS Instructions with RCX >=2^32 May Cause a System Hang

Problem: In IA-32e mode using Intel EM64T-enabled processors, executing a repeating string instruction with the iteration count greater than or equal to 2^32 and a pending event may cause the REP STOS/MOVS instruction to live lock and hang.

Implication: When this erratum occurs, the processor may live lock and result in a system hang. Intel has not observed this erratum with any commercially available software.

Workaround: Do not use strings larger than 4 GB.

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

AA13. A 64-Bit Value of Linear Instruction Pointer (LIP) May be Reported Incorrectly in the Branch Trace Store (BTS) Memory Record or in the Precise Event Based Sampling (PEBS) Memory Record

Problem: On a processor supporting Intel® EM64T,

- If an instruction fetch wraps around the 4G boundary in Compatibility Mode, the 64-bit value of LIP in the BTS memory record will be incorrect (upper 32 bits will be set to FFFFFFFFh when they should be 0).
- If a PEBS event occurs on an instruction whose last byte is at memory location FFFFFFFFh, the 64-bit value of LIP in the PEBS record will be incorrect (upper 32 bits will be set to FFFFFFFFh when they should be 0).

Implication: 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*.

AA14. Access to an Unsupported Address Range in Uniprocessor (UP) or Dual-processor (DP) Systems Supporting Intel® Virtualization Technology May Not Trigger Appropriate Actions

Problem: When using processors supporting Intel® Virtualization Technology and configured as dual- or single-processor-capable (i.e. not multiprocessor-capable), the processor should perform address checks using a maximum physical address width of 36. Instead, these processors will perform address checks using a maximum physical address width of 40.

Implication: Due to this erratum, actions which are normally taken upon detection of an unsupported address may not occur. Software which does not attempt to access unsupported addresses will not experience this erratum.

Workaround: None identified.

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

AA15. VM Exit Due to a MOV from CR8 May Cause an Unexpected Memory Access

Problem: In a system supporting Intel® Virtualization Technology and Intel® Extended Memory 64 Technology, if the "CR8-store exiting" bit in the processor-based VM-execution control field is set and the "use TPR shadow" bit is not set, a MOV from CR8 instruction executed by a Virtual Machine Extensions (VMX) guest that causes a VM exit may generate an unexpected memory access.

Implication: When this erratum occurs, a read access to unexpected address may be issued to the chipset. Subsequent side effects are dependent on chipset operation and may include system hang.

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.*

AA16. The Processor May Incorrectly Respond to Machine Checks during VM Entry/Exit Transitions

Problem: In systems supporting Intel® Virtualization Technology, when machine checks are encountered during VM entry/exit transitions, the processor is expected to respond with a VM exit (if a machine check occurs during VM entry) or abort (if a machine check occurs during VM exit). As a result of this erratum when machine checks occur during VM entry/exit transitions the processor will attempt to service the machine check which may lead to IERR-shutdown or execution of the Machine Check handler, dependent on the CR4.MCE setting.

Implication: The system may end up in the shutdown state if CR4.MCE is not set.

Workaround: None identified.

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

AA17. Power Down Requests May not be Serviced if a Power Down Transition is Interrupted by an In-Target Probe Event in the Presence of a Specific Type of VM Exit

Problem: In a system supporting Intel® Virtualization Technology, the processor may service a pended VM exit prior to completely exiting out of a low power state when the following sequences of events occur:

- Chip-wide power down transition occurs and
- VM exit due to a VMLaunch, VMResume, STI, POPF, POPFD, or IRET instruction is pended and
- Chip-wide power down transition is interrupted by an In-Target Probe event.

Implication: Due to this erratum the processor may not recognize further STPCLK# assertions, TM1, TM2, or Enhanced Intel SpeedStep® technology. 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.*

AA18. VM EXIT Due to TPR shadow Below Threshold May Improperly Set and Cause "Blocking by STI" actions

Problem: In a system supporting Intel® Virtualization Technology and Intel® EM64T, the "blocking by STI" bit of the interruptibility-state field may be saved as 1 rather than 0. This erratum may occur when a STI instruction is executed directly prior to a MOV to CR8 which results in a VM exit due to a reduction of the TPR shadow value below the TPR threshold.

Implication: When this erratum occurs, delivery of an interrupt may be delayed by one instruction.

Workaround: None identified.

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

AA19. Two Correctable L2 Cache Errors in Close Proximity May Cause a System Hang

Problem: If two correctable L2 cache errors are detected in close proximity to each other, a livelock may occur as a result of the processor being unable to resolve this condition.

Implication: When this erratum occurs, the processor may livelock and result in a system hang. Intel has only observed this erratum while injecting cache errors in simulation..

Workaround: None identified.

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

AA20. A VM Exit due to SMI or INIT in Parallel with a Pending FP Exception May Not Correctly Clear the Interruptibility State Bits

Problem: When a pending FP exception is ready to be taken, a VM exit due to SMI or INIT may not clear Blocking by STI and/or Blocking by MOV SS bits correctly in Virtual-Machine Control Structure (VMCS) as expected..

Implication: A VM exit due to SMI or INIT may show incorrect STI and/or MOV SS blocking state in VM-exit Interruptibility field.

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.*

AA21. Processor May Hang with a 25% or Less STPCLK# Duty Cycle

Problem: If a system de-asserts STPCLK# at a 25% or less duty cycle and the processor thermal control circuit (TCC) on-demand clock modulation is active, the processor may hang. This erratum does not occur under the automatic mode of the TCC.

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

Workaround: If use of the on-demand mode of the processor's TCC is desired in conjunction with STPCLK# modulation, then assure that STPCLK# is not asserted at a 25% duty cycle.

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

AA22. Attempting to Use an LDT Entry when the LDTR Has Been Loaded with an Unusable Segment May Cause Unexpected Memory Accesses

Problem: In a system supporting Intel® Virtualization Technology, the processor may incorrectly VM exit under the following conditions:

1. Interrupt-Window-Exiting VM-execution control is set
2. RFLAGS[IF]=1
3. Chipwide Powerdown transition requests occur when the processor is in Wait-For-SIPI or Shutdown states

Implication: Due to this erratum, Interrupt-Window-Exiting VM exits may take the logical processor out of Wait-For-SIPI and Shutdown states. 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.

AA23. Machine Check Exceptions May not Update Last-Exception Record MSRs (LERs)

Problem: The Last-Exception Record MSRs (LERs) may not get updated when Machine Check Exceptions occur

Implication: When this erratum occurs, the LER may not contain information relating to the machine check exception. They will contain information relating to the exception prior to the machine check exception.

Workaround: None identified

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

AA24. VM Entry/Exit Writes to LSTAR/SYSCALL_FLAG MSR's May Cause Incorrect Data to be Written to Bits [63:32]

Problem: Incorrect MSR data in bits [63:32] may be observed in the following two cases:

1. When ECX contains 0xC0000084 and a VM entry/exit writes the IA32_CR_LSTAR MSR (MSR Address 0xC0000082) bits [63:32] of the data may be zeroed
2. When ECX does not contain 0xC0000084 and a VM entry/exit writes the IA32_CR_SYSCALL_FLAG_MASK MSR (MSR Address 0xC0000084) bits [63:32] of the data may not be zeroed

Implication: Bits [63:32] of the affected MSRs may contain the wrong data after a VM exit/entry which loads the affected MSR.

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.

AA25. Writing the Local Vector Table (LVT) when an Interrupt is Pending May Cause an Unexpected Interrupt

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

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

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

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

AA26. At a Bus Ratio of 13:1, RCNT and Address Parity May be Incorrect

Problem: In a system running at the 13:1 bus ratio, RCNT[0] (ADDR# [28], phase b) may report incorrect information.

Implication: RCNT[0] may contain incorrect information and cause address parity machine check errors.

Workaround: Address parity should be disabled and RCNT information should be ignored at the bus ratio of 13:1.

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

AA27. The Execution of a VMPTRLD Instruction May Cause an Unexpected Memory Access

Problem: In a system supporting Intel® Virtualization Technology, executing VMPTRLD may cause a memory access to an address not referenced by the memory operand.

Implication: This erratum may cause unpredictable system behavior including system hang.

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.*

AA28. The Execution of VMPTRLD or VMREAD May Cause an Unexpected Memory Access

Problem: On processors supporting Intel® Virtualization Technology, executing a VMPTRLD or a VMREAD instruction outside of VMX mode may result in a load to an unexpected address.

Implication: This erratum may cause a load to an unexpected memory address.

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.*

AA29. On a “Failed VM-entry” VM Exit, the VMCS Pointer May have Incorrect Value

Problem: On a “failed VM-entry” VM exit, the VMCS pointer may have incorrect value.

Implication: The value of the VMCS pointer may be incorrect and may result in unpredictable behavior after the "failed 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.*

AA30. During an Enhanced HALT or Enhanced Intel SpeedStep® Technology Ratio Transition the System May Hang

Problem: The BNR signal may not function properly during an Enhanced HALT or Enhanced Intel SpeedStep Technology ratio transition.

Implication: The system may hang due to incorrect BNR signaling.

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.*

AA31. L2 Cache ECC Machine Check Errors May be erroneously Reported after an Asynchronous RESET# Assertion

Problem: Machine check status MSRs may incorrectly report the following L2 Cache ECC machine-check errors when cache transactions are in-flight and RESET# is asserted:

- Instruction Fetch Errors (IA32_MC2_STATUS with MCA error code 153)
- L2 Data Write Errors (IA32_MC1_STATUS with MCA error code 145)

Implication: Uncorrected or corrected L2 ECC machine check errors may be erroneously reported. Intel has not observed this erratum on any commercially available system.

Workaround: When a real run-time L2 Cache ECC Machine Check occurs, a corresponding valid error will normally be logged in the IA32_MC0_STATUS register. BIOS may clear IA32_MC2_STATUS and/or IA32_MC1_STATUS for these specific errors when IA32_MC0_STATUS does not have its VAL flag set.

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

AA32. VMCALL to Activate Dual-monitor Treatment of SMIs and SMM Ignores Reserved Bit settings in VM-exit Control Field

Problem: Processors supporting Intel® Virtualization Technology can execute VMCALL from within the Virtual Machine Monitor (VMM) to activate dual-monitor treatment of SMIs and SMM. Due to this erratum, if reserved bits are set to values inconsistent with VMX Capability MSRs, VMCALL may not VMFail.

Implication: VMCALL executed to activate dual-monitor treatment of SMIs and SMM may not VMFail due to incorrect reserved bit settings in VM-Exit control field.

Workaround: Software should ensure that all VMCS reserved bits are set to values consistent with VMX Capability MSRs.

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

AA33. Using 2M/4M Pages When A20M# Is Asserted May Result in Incorrect Address Translations

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

- paging is enabled
- a linear address has bit 20 set
- the address references a large page
- A20M# is enabled

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

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

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

AA34. Writing Shared Unaligned Data that Crosses a Cache Line without Proper Semaphores or Barriers May Expose a Memory Ordering Issue

Problem: Software which is written so that multiple agents can modify the same shared unaligned memory location at the same time may experience a memory ordering issue if multiple loads access this shared data shortly thereafter. Exposure to this problem requires the use of a data write which spans a cache line boundary.

Implication: This erratum may cause loads to be observed out of order. Intel has not observed this erratum with any commercially available software or system.

Workaround: Software should ensure at least one of the following is true when modifying shared data by multiple agents:

- The shared data is aligned
- Proper semaphores or barriers are used in order to prevent concurrent data accesses

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

AA35. The IA32_MC0_STATUS and IA32_MC1_STATUS Overflow Bit is not set when Multiple Un-correctable Machine Check Errors Occur at the Same Time

Problem: When two enabled MC0/MC1 un-correctable machine check errors are detected in the same bank in the same internal clock cycle, the highest priority error will be logged in IA32_MC0_STATUS / IA32_MC1_STATUS register, but the overflow bit may not be set.

Implication: The highest priority error will be logged and signaled if enabled, but the overflow bit in the IA32_MC0_STATUS/ IA32_MC1_STATUS register may not be set.

Workaround: None identified.

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

AA36. 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.*

AA37. Processor May Fault When the Upper 8 Bytes of Segment Selector Is Loaded from a Far Jump through a Call Gate via the Local Descriptor Table

Problem: In IA-32e mode of the Intel EM64T processor, control transfers through a call gate via the Local Descriptor Table (LDT) that uses a 16-byte descriptor, the upper 8-byte access may wrap and access an incorrect descriptor in the LDT. This only occurs on an LDT with a LIMIT>0x10008 with a 16-byte descriptor that has a selector of 0xFFFC.

Implication: In the event this erratum occurs, the upper 8-byte access may wrap and access an incorrect descriptor within the LDT, potentially resulting in a fault or system hang. 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*.

AA38. The Processor May Issue Front Side Bus Transactions up to 6 Clocks after RESET# is Asserted

Problem: The processor may issue transactions beyond the documented 3 Front Side Bus (FSB) clocks and up to 6 FSB clocks after RESET# is asserted in the case of a warm reset. A warm reset is where the chipset asserts RESET# when the system is running.

Implication: The processor may issue transactions up to 6 FSB clocks after the RESET# is asserted

Workaround: None identified.

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

AA39. Front Side Bus Machine Checks May be Reported as a Result of On-Going Transactions during Warm Reset

Problem: Processor Front Side Bus (FSB) protocol/signal integrity machine checks may be reported if the transactions are initiated or in-progress during a warm reset. A warm reset is where the chipset asserts RESET# when the system is running.

Implication: The processor may log FSB protocol/signal integrity machine checks if transactions are allowed to occur during RESET# assertions.

Workaround: BIOS may clear FSB protocol/signal integrity machine checks for systems/chipsets which do not block new transactions during RESET# assertions.

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

AA40. NMI-blocking Information Recorded in VMCS May be Incorrect after a #GP on an IRET Instruction

Problem: In a system supporting Intel® Virtualization Technology, the NMI blocking bit in the Interruption-Information Field in the guest VMCS may be set incorrectly. This erratum will happen if a VMExit occurs for a #GP fault on an IRET instruction due to an EIP that violates the segment limit or is non-canonical.

Implication: If this erratum occurs, monitor software may not be able to handle #GP and then inject an NMI since monitor software does not have information about whether NMIs are blocked in the guest.

Workaround: Monitor software can workaround this bug by avoiding injection of NMI after #GP emulation.

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

AA41. VMLAUNCH/VMRESUME May Not Fail when VMCS is Programmed to Cause VM Exit to Return to a Different Mode

Problem: VMLAUNCH/VMRESUME instructions may not fail if the value of the "host address-space size" VM-exit control differs from the setting of IA32_EFER.LMA.

Implication: Programming the VMCS to allow the monitor to be in different modes prior to VMLAUNCH/VMRESUME and after VM-exit may result in undefined behavior.

Workaround: Software should ensure that "host address-space size" VM-exit control has the same value as IA32_EFER.LMA at the time of VMLAUNCH/VMRESUME.

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

AA42. A Continuous Loop Executing Bus Lock Transactions on One Logical Processor may Prevent Another Logical Processor from Acquiring Resources

Problem: In a system supporting Hyper-Threading Technology, when one hardware thread is in a continuous loop executing bus locks plus other traffic, the other hardware thread may be prevented from acquiring resources to also execute a lock.

Implication: This erratum may cause system hang or unpredictable system behavior. This erratum has not been observed with commercially available software.

Workaround: None identified.

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

AA43. An Unexpected Memory Access May be Issued During Execution of the WRMSR Instruction Under Certain Conditions

Problem: The processor may generate an unexpected memory access during the WRMSR instruction under certain conditions.

Implication: When this erratum occurs, an unexpected read may be issued on FSB. Subsequent side effects are dependent on platform operation and may include a system hang. This erratum has not been observed on any commercial operating system.

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

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

AA44. Combining Some Processors With Intel 945® Chipsets Can Lead to Unpredictable System Behavior

Problem: Some processors with 800 MHz Front Side Bus (FSB), when used in combination with a motherboard based on the Intel 945® chipset, may observe FSB bit errors which may result in unpredictable system behavior.

Implication: Due to this erratum, FSB marginality is observed during processor core to core transactions as well as during read transactions driven by the Memory Controller Hub (MCH) leading to unpredictable system behavior.

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

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

AA45. A VM Exit Occuring in IA-32e Mode May Not Produce a VMX Abort When Expected

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

Implication: A VM Exit will occur when a VMX Abort was expected.

Workaround: An SMM VMM should always set the "IA-32e guest" VM-entry control in the SMM VMCS to be the value that was in the LMA bit (IA32_EFER.LMA.LMA[bit 10]) in the IA32_EFER MSR (C0000080H) at the time of the last SMM VM exit. If this guideline is followed, that value will be 1 only if the "host address-space size" VM-exit control is 1 in the executive VMCS.

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

§

Specification Changes

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

- *Intel® Pentium® D Processor 900 Sequence and Intel® Pentium® Processor Extreme Edition 955, 965 Datasheet*

All Specification Changes will be incorporated into a future version of the appropriate Pentium D processor 900 sequence and Pentium processor Extreme Edition 955, 965 documentation.

Δ Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Over time processor numbers will increment based on changes in clock, speed, cache, FSB, or other features, and increments are not intended to represent proportional or quantitative increases in any particular feature. Current roadmap processor number progression is not necessarily representative of future roadmaps. See www.intel.com/products/processor_number for details.

§

Specification Clarifications

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

- *Intel® Pentium® D Processor 900 Sequence and Intel® Pentium® Processor Extreme Edition 955, 965 Datasheet*

All Specification Clarifications will be incorporated into a future version of the appropriate Pentium D processor 900 sequence and Pentium processor Extreme Edition 955, 965 documentation.

§

Documentation Changes

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

- *Intel® Pentium® D Processor 900 Sequence and Intel® Pentium® Processor Extreme Edition 955, 965 Datasheet*

All Documentation Changes will be incorporated into a future version of the appropriate Pentium D processor 900 sequence and Pentium processor Extreme Edition 955, 965 documentation.

Note: Documentation changes for Intel® 64 and IA-32 Architecture Software Developer's Manual volumes 1, 2A, 2B, 3A, and 3B will be posted in a separate document Intel® 64 and IA-32 Architecture Software Developer's Manual Documentation Changes. Follow the link below to become familiar with this file.

http://developer.intel.com/design/pentium4/specupdt/252046.htm

There are no documentation changes in this Specification Update revision.