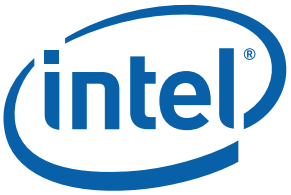


# Intel® Xeon® Processor E7-8800/4800/2800 Product Families

## Specification Update

***August 2015***

Reference Number: 325122-024

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

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

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

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

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

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

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

Intel is a sponsor and member of the Benchmark XPRT Development Community, and was the major developer of the XPRT family of benchmarks. Principled Technologies is the publisher of the XPRT family of benchmarks. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases.

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

Intel, the Intel logo, and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

# Contents



§

# Revision History

| Revision | Description | Date |
|---|---|---|
| -001 | Public Release | April 2011 |
| -002 | Added BP33 errata | April 2011 |
| -003 | Added BP34, BP35 errata | May 2011 |
| -004 | Added erratum BP36 | July 2011 |
| -005 | Added erratum BP37 | August 2011 |
| -006 | Added erratum BP38 | September 2011 |
| -007 | Added erratum BP39 | October 2011 |
| -008 | Added erratum AS2. Updated Tables 2 and 3 | December 2011 |
| -009 | Added errata BP40 and BP41 | February 2012 |
| -010 | Added erratum BP42 | April 2012 |
| -011 | Added errata BP43 through BP47 | May 2012 |
| -012 | Added errata BP48 and BP49 | June 2012 |
| -013 | Added specification clarification SC1 | August 2012 |
| -014 | Added erratum BP50 | September 2012 |
| -015 | Added errata BP51 and BP52 | December 2012 |
| -016 | Added documentation change DC1 | January 2013 |
| -017 | Added erratum BP53 | May 2013 |
| -018 | Added errata BP54, BP55, and BP56 | June 2013 |
| -019 | Added erratum BP57 | August 2013 |
| -020 | Added erratum BP58 and specification change SCh1 | March 2014 |
| -021 | Added erratum BP59 | November 2014 |
| -022 | Updated erratum BP53 | February 2015 |
| -023 | Added erratum BP60 | March 2015 |
| -024 | Added erratum BP61 | August 2015 |

# Preface

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

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

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

## Affected Documents

| Document Title | Document Number/ Location |
|---|---|
| *Intel® Xeon® Processor E7-8800/4800/2800 Product Families Datasheet Volume 1* | 325119 |
| *Intel® Xeon® Processor E7-8800/4800/2800 Product Families Datasheet Volume 2* | 325120 |

## Related Documents

| Document Title | Document Number/ Location |
|---|---|
| *Intel® 64 and IA-32 Architectures Software Developer's Manual*<br>• Volume 1: Basic Architecture<br>• Volume 2A: Instruction Set Reference Manual A-M<br>• Volume 2B: Instruction Set Reference Manual N-Z<br>• Volume 3A: System Programming Guide<br>• Volume 3B: System Programming Guide | http://www.intel.com/products/processor/manuals/index.htm |
| *Intel® 64 and IA-32 Intel Architectures Optimization Reference Manual* | 248966 |
| *Intel® 64 and IA-32 Architectures Software Developer's Manual Documentation Changes* | 252046 |

## Nomenclature

**Errata** are design defects or errors. These may cause the Intel® Xeon® Processor E7-8800/4800/2800 Product Families 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.

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

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

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

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

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

# Summary Tables of Changes

The following tables indicate the errata, specification changes, specification clarifications, or documentation changes which apply to the Intel® Xeon® Processor E7-8800/4800/2800 Product Families. Intel may fix some of the errata in a future stepping of the component, and account for the other outstanding issues through documentation or specification changes as noted. These tables uses the following notations:

## Codes Used in Summary Tables

### Stepping

| | |
|---|---|
| X: | Errata exists in the stepping indicated. Specification Change or Clarification that applies to this stepping. |
| (No mark) or (Blank box): | This erratum is fixed in listed stepping or specification change does not apply to listed stepping. |

### Page

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

### Status

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

### Row

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

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:

## Table 1. Errata Table (Sheet 1 of 2)

| Number | Stepping | Status | Description |
|---|---|---|---|
| | A-2 | | |
| BP1. | X | No Fix | Intel® Interconnect BIST (Intel® IBIST) Does Not Work in Intel® QuickPath Interconnect (Intel® QPI) in Slow Mode |
| BP2. | X | No Fix | Retraining Parameter Negotiation is Not Implemented for Intel® QPI |
| BP3. | X | No Fix | Intel® IBIST Slave Ignores Loop Count Values Sent by Master on Intel® QPI |
| BP4. | X | No Fix | System Hangs when Skipping Stop Req2 and Start Req1 Messages in Quiesce/Lock Sequence |
| BP5. | X | No Fix | Integrated Memory Controller Signals Spurious CMCI when Home Agent Failover Count Saturation Occurs |
| BP6. | X | No Fix | Memory Controller Does Not Set S Bit for Uncorrectable Error Followed by Software Recoverable Error |
| BP7. | X | No Fix | MCi_STATUS S Bit Not Set for LLC Software Recoverable Errors |
| BP8. | X | No Fix | Correctable SB CRC Error May be Propagated to an Uncorrected ECC Error |
| BP9. | X | No Fix | Memory Controller Patrol Scrub Ceases to Function with CRC Errors and the IMT31 Reclaim Feature Enabled |
| BP10. | X | No Fix | Electrically Idle Intel® SMI and Intel® QPI Lanes May Deliver Data that May Look Like Deskew Headers |
| BP11. | X | No Fix | A Sequence of Instruction Fetches and Snoops to Locked Cache Lines May Cause Processor to Hang |
| BP12. | X | No Fix | Writing to Unimplemented Bits of UU_CR_U_MSR_PMON_EVNT_SEL MSR does Not Result in #GP Fault |
| BP13. | X | No Fix | Mixed Rank Size Memory Configurations May Cause a Missing Refresh Event |
| BP14. | X | No Fix | Mirror Slave May Deliver Incorrect Data when a Read to the Mirror Master Completes Before the Write-back from the IOH |
| BP15. | X | No Fix | UU_CR_U_MSR_PMON_GLOL_OVF_CTL MSR Does Not Follow RW1C Access Method |
| BP16. | X | No Fix | HNID Field is Incorrect for CMP Messages From PrefetchHint |
| BP17. | X | No Fix | Page Fault May Occur When Logical Processor Transitions From C6 State to C0 State |
| BP18. | X | No Fix | In DAS Enabled Mode a System Hang May Occur During Memory Intensive Workloads |
| BP19. | X | No Fix | Bit [8] of IA32_APIC_BASE register Inadvertently Set to 1 for Core 9 |
| BP20. | X | No Fix | Quad Rank DIMMs With CKE Low Enabled in Open/Adaptive Page Mode May Return Incorrect Data |
| BP21. | X | No Fix | System Configuration Controller Misaligned Error May Result in a System Hang |
| BP22. | X | No Fix | Recoverable Errors Signaled From Intel® QPI or Intel® SMI Port to the System Configuration Controller May Get Lost if the Ports are Disabled |
| BP23. | X | No Fix | Executing The WAKEUP Leaf of The GETSEC Instruction Multiple Times May Lead to a Machine Check Error |
| BP24. | X | No Fix | CKE-Lo Feature Can Not be Disabled When Memory Controller Transactions are Active |
| BP25. | X | No Fix | Executing The Intel TXT GETSEC SENTER Instruction Leaf May Lead to a Machine Check Error |
| BP26. | X | No Fix | Task Switch to a TSS With an Inaccessible LDTR Descriptor May Cause Unexpected Faults |
| BP27. | X | No Fix | An Intel® QPI Link Layer Retry Quickly Followed by an Intel® QPI Physical Layer Reset May Cause an MCE |
| BP28. | X | No Fix | LLC Arrays May have Incorrect Values after Warm Reset when Memory BIST is Disabled |
| BP29. | X | No Fix | VM Entries that Return from SMM May Incorrectly Write to the SMRR Protected Region |
| BP30. | X | No Fix | System Quiesce Events Initiated While Power Events are In Progress May Cause System Hangs |
| BP31. | X | No Fix | Uncorrected Memory Error Detected by a Memory Patrol Scrub With SMI Generated by Other Memory Controllers May Cause MCE/System Management Interrupt Race Condition |
| BP32. | X | No Fix | Broken trace to either the P or the N lane of the Intel® SMI forwarded clock differential pair may result in loss of forwarded clock but not always lead to clock lane failover. |

### Table 1. Errata Table (Sheet 2 of 2)

| Number | Stepping | Status | Description |
|---|---|---|---|
| | A-2 | | |
| BP33. | X | No Fix | Package C3/C6 with Memory Self-refresh Enabled May Cause False Error Logging |
| BP34. | X | No Fix | Performance Monitor WOKEN Event May Under Count |
| BP35. | X | No Fix | PECI Command Average Temperature Read does not report correct Temperature |
| BP36. | X | No Fix | Intel® QPI Initialization May Cause a CATERR During Power-on Reset |
| BP37. | X | No Fix | EOI Transaction May Not be Sent if Software Enters Core C6 During an Interrupt Service Routine |
| BP38. | X | No Fix | A First Level Data Cache Parity Error May Result in Unexpected Behavior |
| BP39. | X | No Fix | An Unexpected Page Fault or EPT Violation May Occur After Another Logical Processor Creates a Valid Translation for a Page |
| BP40. | X | No Fix | A Page Fault May Not be Generated When the PS bit is set to "1" in a PML4E or PDPTE |
| BP41. | X | No Fix | IO_SMI Indication in SMRAM State Save Area May be Set Incorrectly |
| BP42. | X | No Fix | Writing an Illegal Vector to the IA32_X2APIC_SELF_IPI MSR Will Hang the Processor |
| BP43. | X | No Fix | Successive Fixed Counter Overflows May be Discarded |
| BP44. | X | No Fix | VM Exits Due to "NMI-Window Exiting" May Not Occur Following a VM Entry to the Shutdown State |
| BP45. | X | No Fix | Execution of INVVPID Outside 64-Bit Mode Cannot Invalidate Translations For 64-Bit Linear Addresses |
| BP46. | X | No Fix | A Combination of Data Accesses That Are Split Across Cacheline Boundaries May Lead to a Processor Hang |
| BP47. | X | No Fix | A Load May Appear to be Ordered Before an Earlier Locked Instruction |
| BP48. | X | No Fix | VMRESUME May Omit Check of Revision Identifier of Linked VMCS |
| BP49. | X | No Fix | APIC Timer Interrupts May be Lost During Core C3 |
| BP50. | X | No Fix | MCI_ADDR May be Incorrect For Cache Parity Errors |
| BP51. | X | No Fix | CR0.CD Is Ignored in VMX Operation |
| BP52. | 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 |
| BP53. | X | No Fix | The Corrected Error Count Overflow Bit in IA32_ MC0_STATUS is Not Updated When The UC Bit is Set |
| BP54. | X | No Fix | The Upper 32 Bits of CR3 May be Incorrectly Used With 32-Bit Paging |
| BP55. | X | No Fix | EPT Violations May Report Bits 11:0 of Guest Linear Address Incorrectly |
| BP56. | X | No Fix | IA32_VMX_VMCS_ENUM MSR (48AH) Does Not Properly Report The Highest Index Value Used For VMCS Encoding |
| BP57. | X | No Fix | Virtual-APIC Page Accesses With 32-Bit PAE Paging May Cause a System Crash |
| BP58. | X | No Fix | VM Exit May Set IA32_EFER.NXE When IA32_MISC_ENABLE Bit 34 is Set to 1 |
| BP59. | X | No Fix | Performance Monitor Counter MEM_INST_RETIRED.STORES May Count Higher than Expected |
| BP60. | X | No Fix | Pending x87 FPU Exceptions (#MF) May be Signaled Earlier Than Expected |
| BP61. | X | No Fix | An IRET Instruction That Results in a Task Switch Does Not Serialize The Processor |

## Specification Changes

| Number | SPECIFICATION CHANGES |
|---|---|
| SCh1. | Correction to Product Families Features |

## Specification Clarifications

| Number | SPECIFICATION CLARIFICATIONS |
|---|---|
| SC1. | Package C3/C6 Memory Self-Refresh Error Handling |

## Document Changes

| Number | DOCUMENT CHANGES |
|---|---|
| DC1. | On-Demand Clock Modulation Feature Clarification |

# Identification Information

## Component Identification via Programming Interface

The Intel® Xeon® Processor E7-8800/4800/2800 Product Families stepping can be identified by the following register contents:

| Reserved | Extended Family[1] | Extended Model[2] | Reserved | Processor Type[3] | Family Code[4] | Model Number[5] | Stepping ID[6] |
|---|---|---|---|---|---|---|---|
| 31:28 | 27:20 | 19:16 | 15:14 | 13:12 | 11:8 | 7:4 | 3:0 |
| | 00000000b | 0010b | | 00b | 0110 | 1111b | 0000b |

***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® Pro, Pentium® 4, Intel® Core™ processor family or Intel® Core™ i7 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 Processor Type, specified in bits [13:12] indicates whether the processor is an original OEM processor, an OverDrive processor, or a dual processor (capable of being used in a dual processor system).
4. 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.
5. 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.
6. The Stepping ID in bits [3:0] indicates the revision number of that model. See Table 2 for the processor stepping ID number in the CPUID information.

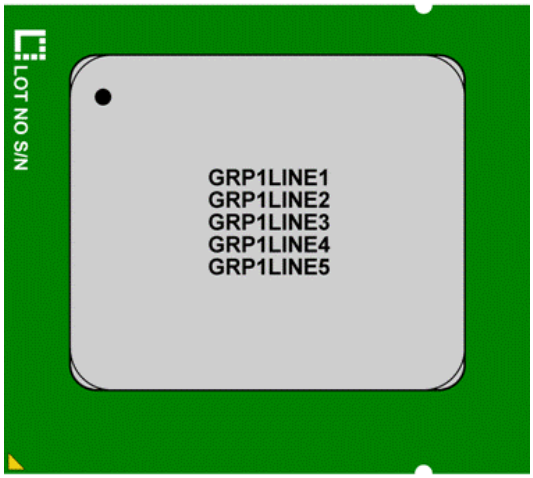
When EAX is initialized to a value of '1', the CPUID instruction returns the *Extended Family, Extended Model, Processor Type, Family Code, Model Number and Stepping ID* value in the EAX register. Note that the EDX processor signature value after reset is equivalent to the processor signature output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

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

## Component Marking Information

Intel® Xeon® Processor E7-8800/4800/2800 Product Families can be identified by the following component markings:

**Figure 1. Processor Top-Side Marking (Example)**




| Legend: | Mark Text (Production Mark): |
|---|---|
| GRP1LINE1: | INTEL{M}{C}'YY PROC# |
| GRP1LINE2: | Intel® Xeon® |
| GRP1LINE3: | SSPEC XXXXX |
| GRP1LINE4: | SPEED/CACHE/INTC |
| GRP1LINE5: | {FPO} {e4} |

YY = Year
PROC# = Processor Number
xxxxx = Country of Origin
INTC = Interconnect Speed (Intel® QPI)
Factory Information

**Table 2. Intel® Xeon® Processor E7-8800/4800/2800 Product Families Identification**

| S-Spec Number | Stepping | CPUID | Core Frequency (GHz) / Intel® QuickPath Interconnect (GT/s) / Intel® SMI (GT/s) | Number of Cores | Cache Size (MB) | Series |
|---|---|---|---|---|---|---|
| SLC3E | A-2 | 000206F2h | 2.4 GHz/6.4 GT/s/6.4 GT/s | 10 | 30 MB | E7-8870 |
| SLC3T | A-2 | 000206F2h | 2.4 GHz/6.4 GT/s/6.4 GT/s | 10 | 30 MB | E7-4870 |
| SLC3U | A-2 | 000206F2h | 2.4 GHz/6.4 GT/s/6.4 GT/s | 10 | 30 MB | E7-2870 |
| SLC3F | A-2 | 000206F2h | 2.26 GHz/6.4 GT/s/6.4 GT/s | 10 | 24 MB | E7-8860 |
| SLC3S | A-2 | 000206F2h | 2.26 GHz/6.4 GT/s/6.4 GT/s | 10 | 24 MB | E7-4860 |
| SLC3H | A-2 | 000206F2h | 2.26 GHz/6.4 GT/s/6.4 GT/s | 10 | 24 MB | E7-2860 |
| SLC3D | A-2 | 000206F2h | 2.0 GHz/6.4 GT/s/6.4 GT/s | 10 | 24 MB | E7-8850 |
| SLC3V | A-2 | 000206F2h | 2.0 GHz/6.4 GT/s/6.4 GT/s | 10 | 24 MB | E7-4850 |
| SLC3W | A-2 | 000206F2h | 2.0 GHz/6.4 GT/s/6.4 GT/s | 10 | 24 MB | E7-2850 |
| SLC3K | A-2 | 000206F2h | 2.13 GHz/6.4 GT/s/6.4 GT/s | 8 | 24 MB | E7-8830 |
| SLC3Q | A-2 | 000206F2h | 2.13 GHz/6.4 GT/s/6.4 GT/s | 8 | 24 MB | E7-4830 |
| SLC3J | A-2 | 000206F2h | 2.13 GHz/6.4 GT/s/6.4 GT/s | 8 | 24 MB | E7-2830 |
| SLC3P | A-2 | 000206F2h | 2.13 GHz/6.4 GT/s/6.4 GT/s | 10 | 30 MB | E7-8867L |
| SLC3N | A-2 | 000206F2h | 2.66 GHz/6.4 GT/s/6.4 GT/s | 8 | 24 MB | E7-8837 |
| SLC3M | A-2 | 000206F2h | 1.73 GHz/4.8 GT/s/4.8 GT/s | 6 | 18 MB | E7-2803 |
| SLC3G | A-2 | 000206F2h | 2.0 GHz/5.86 GT/s/5.86 GT/s | 8 | 18 MB | E7-4820 |
| SLC3R | A-2 | 000206F2h | 2.0 GHz/5.86 GT/s/5.86 GT/s | 8 | 18 MB | E7-2820 |
| SLC3L | A-2 | 000206F2h | 1.86 GHz/4.8 GT/s/4.8 GT/s | 6 | 18 MB | E7-4807 |

## Mixing Processor Within MP Platforms

Intel supports multiprocessor (MP) configurations consisting of processors:

1. From the same power optimization segment.
2. That support the same maximum Intel® QuickPath Interconnect (Intel® QPI) and DDR3 memory speeds.
3. That share symmetry across physical packages with respect to the number of logical processors per package, number of cores per package, number of Intel® QPI interfaces, and cache topology.
4. That have identical Extended Family, Extended Model, Processor Type, Family Code, and Model Number as indicated by the function 1 of the CPUID instruction.

***Note:*** Connected processors must operate with the same Intel® QPI and core frequency.

While Intel does nothing to prevent processors from operating together, some combinations may not be supported due to limited validation, which may result in uncharacterized errata. Coupling this fact with the large number of Intel® Xeon® Processor E7-8800/4800/2800 Product Families attributes, the following population rules and stepping matrix have been developed to define supported configurations.

1. Processors must be of the same power-optimization segment. This ensures processors include the same maximum Intel® QPI and cache sizes.
2. Processors must operate at the same core frequency. Note: Processors within the same power-optimization segment supporting different maximum core frequencies (for example, a 2.26 GHz / 130 W and 2.00 GHz / 130 W) can be operated within a system. However, both must operated at the highest frequency rating commonly supported. Mixing components operating at different internal clock frequencies is not supported and will not be validated by Intel.
3. Processors must share symmetry across physical packages with respect to the number of logical processors per package, number of Intel® QPI interfaces, and cache topology.
4. Mixing dissimilar steppings is only supported with processors that have identical Extended Family, Extended Model, Processor type, Family Code, and Model Number as indicated by the function 1 of the CPUID instruction. Mixing processors of different steppings but the same model (as per CPUID instruction) is supported. Details regarding the CPUID instruction are provided in the *AP-487,* Intel® Processor Identification and the CPUID Instruction application note and *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A.*
5. After ANDing the feature flag and extended feature flag from the installed processors, any processor whose set of feature flags exactly matches the ANDed feature flags can be selected by the BIOS as the BSP. If no processor exactly matches the ANDed feature flag values, then the processors with the numerically lower CPUID should be selected as the BSP.
6. Intel requires that the processor microcode update be loaded on each processor operating within the system. Any processor that does not have the proper microcode update loaded is considered by Intel to be operating out of specification.
7. The workarounds identified in this, and subsequent specification updates, must properly applied to each processor in the system. Certain errata are specific to the multiprocessor environment. Errata for all processor steppings will affect system performance if not properly worked around.
8. Customers are fully responsible for the validation of their system configurations.

# Intel® Trusted Execution Technology Authenticated Control Modules

Platforms supporting Intel® Trusted Execution Technology (Intel® TXT) must ship with authenticated control modules, software binaries used to establish a root of trust.

BIOS launches the BIOS ACM (authenticated control module) to establish a static root of trust at power-on. The measured launch environment launches the SINIT ACM to establish a dynamic root of trust at MLE (Measured Launch Event) launch.

**Table 3. Intel® Xeon® Processor E7-8800/4800/2800 Product Families BIOS ACM Releases**

| Version | Release Date | Stepping | Signature |
|---|---|---|---|
| BIOS ACM 1.0 | 11/2010 | A-2 | Production |
| BIOS ACM 1.1 | 3/2011 | A-2 | Production |
| BIOS ACM 1.2 | 10/2011 | A-2 | Production |

**Table 4. Intel® Xeon® Processor E7-8800/4800/2800 Product Families SINIT ACM Releases**

| Version | Release Date | Stepping | Signature |
|---|---|---|---|
| SINIT ACM 1.0 | 3/2011 | A-2 | Production |
| SINIT ACM 1.1 | 10/2011 | A-2 | Production |

# Intel® Xeon® Processor E7-8800/4800/2800 Product Families BIOS ACM Errata Summary

**Table 5. Intel® Xeon® Processor E7-8800/4800/2800 Product Families BIOS ACM Errata Table**

| Number | Release | | Status | Description |
|---|---|---|---|---|
| | 1.0 | 1.1 | | |
| AC1 | X | | Fixed | BIOS ACM Exit INIT (LockConfig) Call May Fail on Certain IOH Bus Configurations |

# Intel® Xeon® Processor E7-8800/4800/2800 Product Families SINIT ACM Errata Summary

**Table 6. Intel Xeon Processor E7-8800/4800/2800 Product Families SINIT ACM Errata Table**

| Number | Release | | Status | Description |
|---|---|---|---|---|
| | 1.0 | 1.1 | | |
| AS1 | X | X | No Fix | TXT.ERRORCODE TPM Command Return Code And Launch Control Policy List Index And Minor Code Are Not Reported Correctly. |
| AS2 | X | | Fixed | SINIT Buffer Overflow Vulnerability |

# Errata

## BP1. Intel® Interconnect BIST (Intel® IBIST) Does Not Work in Intel® QuickPath Interconnect (Intel® QPI) in Slow Mode

Problem: The Intel IBIST (Interconnect Built-in Self Test) does not work in the Intel® QuickPath Interconnect (Intel® QPI) slow mode and only works at operational speed.

Implication: The Intel IBIST does not work in Intel® QPI slow mode.

Workaround: Do not run the Intel IBIST in slow mode.

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

## BP2. Retraining Parameter Negotiation is Not Implemented for Intel® QPI

Problem: The Intel® QPI specification states that the physical layer initialization process needs to negotiate retraining parameters with a remote agent. The protocol is that agents should first exchange their respective retraining interval and duration as part of the link initialization flow. Then, each agent should compare the local and remote values and choose common values by selecting the shortest interval and the longest duration. This erratum is conveying that the described negotiation feature is not implemented in the processor.

Implication: The processor does not perform the hardware based retraining parameter negotiation.

Workaround: BIOS will need to perform the necessary computations to determine the proper parameters and program them into the processor.

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

## BP3. Intel® IBIST Slave Ignores Loop Count Values Sent by Master on Intel® QPI

Problem: During Intel IBIST (Interconnect Built-in Self Test) loopback, one agent is the master agent while the other is the slave agent on Intel® QPI. The slave should extract an Intel IBIST loop count from the training sequence sent by the master, and use this count to time its stay in the Loopck.Pattern state before returning to Loopck.Marker state. While the processor is operating as a slave, it does not extract this loop count and times its stay in the Loopck.Pattern state based on its locally programmed loop count.

Implication: When operating as an Intel IBIST slave, the processor ignores the loop count values sent by the master.

Workaround: Software needs to program the same loop count into the master and the slave.

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

## BP4. System Hangs when Skipping Stop Req2 and Start Req1 Messages in Quiesce/Lock Sequence

Problem: Quiesce master skipping the StopReq2 and the StartReq1 Intel® QPI messages in the lock sequence will result in a system hang.

Implication: Due to this erratum, quiesce master lock flows with no StopReq2 and StartReq1 messages will cause a system hang.

Workaround: StopReq2 and StartReq1 messages should not be considered optional by quiesce master and must be sent to the processor as part of any lock flow.

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

### BP5. Integrated Memory Controller Signals Spurious CMCI when Home Agent Failover Count Saturation Occurs

Problem: When home agent failover count saturation occurs, the memory controller signals a spurious CMCI (Corrected Machine Check Interrupt) without logging an error. Failover count saturation is not an error and a CMCI should not be issued.

Implication: Due to this erratum, software receives a CMCI with no error logged.

Workaround: None identified.

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

### BP6. Memory Controller Does Not Set S Bit for Uncorrectable Error Followed by Software Recoverable Error

Problem: If an uncorrectable memory controller error is followed by a software recoverable error, the memory controller will not set the S (Signaling flag) bit of the MCi_STATUS to indicate that a software recoverable error occurred.

Implication: Due to this erratum, the MCi_STATUS of the memory controller will have the fields Valid=1, UC=1, PCC=0, OVER=1 and S=0 logged. When the MCA handler comes in, it ignores the MCi_STATUS since S=0; and the MCA is treated as a spurious MCA.

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*.

### BP7. MCi_STATUS S Bit Not Set for LLC Software Recoverable Errors

Problem: When an explicit LLC (Last Level Cache) write-back software recoverable error is detected while there is already a poison error in the MCi_STATUS register, a machine check is signaled but the MCi_STATUS.S (Signaling Flag) bit is not set. In this case the MCi_STATUS.PCC (Processor Context Corrupt) bit and the S bit are both 0. As a result, the machine check handler assumes this to be a spurious error.

Implication: If there is already a poison error in the MCi_STATUS register and an LLC recoverable error is then logged the MCA handler may assume this to be a spurious 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*.

### BP8. Correctable SB CRC Error May be Propagated to an Uncorrected ECC Error

Problem: Due to the processor not having a mechanism to detect incorrect alert frames, correctable SB (South Bound) CRC Error may be propagated to an uncorrected ECC error.

Implication: An incorrect alert frame will not be detected by the processor. In most cases there is no issue, due to the memory buffer issuing a series of alert frames. In a specific case where an SB Intel® Scalable Memory Interconnect (Intel® SMI) CRC error (transient or persistent) is detected and the NB (North Bound) Alert frame responding to this error is also corrupted by an error, the original packet may not be reissued. However, since the memory controller uses two Intel® SMI channels in lockstep for each cache line access, on a future read if one channel was affected by this issue the other would return valid data. Due to this erratum, the correctable SB CRC error may get propagated to be a detected but uncorrected ECC error. Intel has not observed this erratum on any commercially available system.

Workaround: None identified.

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

### BP9. Memory Controller Patrol Scrub Ceases to Function with CRC Errors and the IMT31 Reclaim Feature Enabled

Problem: The processor does not fully implement the protocol in the Memory Controller-Home Agent for sharing the IMT31 (In-flight Memory Table) entry resulting in a patrol scrub deadlock. This issue can occur whenever the Error Flow State is invoked in response to CRC errors or hardware injected periodic ZQCAL (ZQ Calibration).

Implication: Patrol scrub may not function with CRC errors and the IMT31 reclaim feature enabled.

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

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

### BP10. Electrically Idle Intel® SMI and Intel® QPI Lanes May Deliver Data that May Look Like Deskew Headers

Problem: Intel® SMI or Intel® QPI lanes that are not physically connected on the board, or have become unconnected, may result in a deskew failure. A deskew failure or environmental issue may lead to Intel® SMI link transitioning into a Lane Failover Mode.

Implication: Improper deskew headers may be observed if the Intel® SMI lane of a port is not physically connected. The Intel® SMI link may transition into Lane Fail-over mode.

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*.

### BP11. A Sequence of Instruction Fetches and Snoops to Locked Cache Lines May Cause Processor to Hang

Problem: During a sequence of instruction fetches with specific address relationships to other system traffic a snoop beat pattern that includes snoops to locked cache lines may become established which could cause the processor to hang.

Implication: The processor may hang under a set of conditions involving instruction fetches, and snoops to locked cache lines.

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*.

### BP12. Writing to Unimplemented Bits of UU_CR_U_MSR_PMON_EVNT_SEL MSR does Not Result in #GP Fault

Problem: The bits [31:23,17,15:8] in UU_CR_U_MSR_PMON_EVNT_SEL MSR (C10H) are not implemented on the processor and are marked as reserved. Due to this erratum writing 1's to these bits does not generate a #GP (General Protection Fault) as expected.

Implication: Writing 1's to the unimplemented bits in UU_CR_U_MSR_PMON_EVNT_SEL MSR does not result in a #GP fault.

Workaround: None identified.

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

### BP13. Mixed Rank Size Memory Configurations May Cause a Missing Refresh Event

Problem: When using DIMMs of different rank sizes on the same memory channel, a refresh may be missed when a write command to a memory rank is blocked by sustained reads to another memory rank. This erratum has been seen only in a synthetic testing environment. Intel has not observed this erratum with any commercially available software.

Implication: A missing refresh may cause the refresh rate to be lower than the programmed value.

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*.

### BP14. Mirror Slave May Deliver Incorrect Data when a Read to the Mirror Master Completes Before the Write-back from the IOH

Problem: A read from the mirror master may complete before the write-back from the IOH completes. This will result in the IOH write-data not being immediately visible and can lead to the IOH write-data never becoming visible. In the case of a RdInvOwn transaction, the reading caching agent will take ownership which can then overwrite the IOH data. This is especially visible in false-sharing of cache lines which involve the IOH.

Implication: Due to this erratum, correct data is not delivered by the mirror slave. This erratum only occurs during mirror failover.

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*.

### BP15. UU_CR_U_MSR_PMON_GLOL_OVF_CTL MSR Does Not Follow RW1C Access Method

Problem: The UU_CR_U_MSR_PMON_GLOL_OVF_CTL MSR (C02H) is access type RW1C (Read Write 1 Clear) and when written with 1's should clear the corresponding bit in the UU_CR_U_MSR_PMON_ GLOL_STATUS MSR (C01H). Due to this erratum, a read of the UU_CR_U_MSR_PMON_GLOL_OVF_CTL MSR does not return zeros however a read of the UU_CR_U_MSR_PMON_ GLOL_STATUS MSR will show appropriate clearing.

Implication: The UU_CR_U_MSR_PMON_GLOL_OVF_CTL MSR does not return zeros on a read.

Workaround: None identified.

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

### BP16. HNID Field is Incorrect for CMP Messages From PrefetchHint

Problem: The HNID (Home Node ID) field used in the PMON (Performance Monitoring) match/mask is incorrect for CMP (complete) messages from PrefetchHint. The same incorrect HNID is logged in the event of an error condition on a CMP for a PrefetchHint in the caching agent. The logging of the RNID (Requester Node ID) in the error logs for NDR (Non Data Response) messages is incorrect and impacts the caching agent system bound errors.

Implication: There are two implications:

1. Incorrect HNID is filtered or matched for CMPs using the caching agent PMON match/mask.
2. The incorrect RNID will be logged only for errors on NDR messages.

Workaround: There are two potential workarounds:

1. The Intel® QPI performance monitor match/mask can be used to count different types of CMP messages from the caching agent.
2. Ignore the RNID field for NDR system bound messages.

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

### BP17. Page Fault May Occur When Logical Processor Transitions From C6 State to C0 State

Problem: An unexpected Page Fault may occur when a logical processor transitions from C6 to C0, with IA-32e mode enabled.

Implication: Due to this erratum, an unexpected Page Fault may occur during stress testing when the processor core transitions from C6 to C0.

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*.

### BP18. In DAS Enabled Mode a System Hang May Occur During Memory Intensive Workloads

Problem: DAS (Directory Assisted Snoopy) enabled systems may hang with home agent IMT (In-Flight Memory Table) state transition error during stress test.

Implication: This erratum results in home agent timeout or IMT state transition/IMT parity error causing a system hang.

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

Workaround: A BIOS workaround has been identified. Please refer to reference code version 2.0 or later and release notes.

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

### BP19. Bit [8] of IA32_APIC_BASE register Inadvertently Set to 1 for Core 9

Problem: The processor reset flow incorrectly sets BSP bit [8] to 1 in the IA32_APIC_BASE register for Core 9 (of 10 cores).

Implication: When BIOS wakes up Application Processors (APs) using INIT-SIPI-SIPI, BIOS may identify more than one Boot Strap Processor (BSP). This may lead to unpredictable system behavior.

Workaround: Clear bit [8] in the IA32_APIC_BASE register prior to the BIOS MP Initialization routine.

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

### BP20. Quad Rank DIMMs With CKE Low Enabled in Open/Adaptive Page Mode May Return Incorrect Data

Problem: Memory reads may return incorrect data with CKE (Clock Enabled) Low enabled while running with homogeneous Quad Rank DIMMs in Open Page or Adaptive Page Mode.

Implication: System memory may return incorrect data in this configuration.

Workaround: Disable CKE Low if supporting Quad Rank DIMMs in Open or Adaptive Page Mode.

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

### BP21. System Configuration Controller Misaligned Error May Result in a System Hang

Problem: Under certain conditions the system configuration controller may not correctly handle NcRd (Non-Coherent Read) packets which may result in a misaligned uncorrectable error, Machine Check Exception or system hang

Implication: The system configuration controller incorrectly signals an uncorrectable error resulting in a system hang.

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

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

### BP22. Recoverable Errors Signaled From Intel® QPI or Intel® SMI Port to the System Configuration Controller May Get Lost if the Ports are Disabled

Problem: Each Intel® QPI and Intel® SMI physical layer port may be configured through its PBOXERRMASK register (Device:0x14, Function:0x2, Offset:0x68) to generate RAS recoverable error signals in any of the four situations: initialization failure, width reduction (Intel® QPI) or lane failover (Intel® SMI), drift buffer alarm, or latency buffer

rollover. When generated, the error signal is sent to the system configuration controller where it is processed into a system management interrupt (SMI).

Under specific conditions, a RAS recoverable error signal is generated and logged in a physical layer port, but the interrupt is not generated. More specifically, the error signal is lost on the way from the port to the system configuration controller.

The problem arises when the error signal passes through a port that has been disabled. Each physical layer port has its own internal clock generator. When a port is disabled, its clock generator is off, and the error signal cannot propagate through that port.

Implication: If any physical layer ports are configured to signal errors of the RAS recoverable type, then depending on the pattern of disabled ports, the errors may be logged properly in the physical layer port, but a matching system management interrupt may not occur. Fatal error signals are not affected; they will always be transmitted successfully.

Workaround: Do not disable physical layer ports, or if they have been disabled then re-enable them, such that ports that may generate RAS recoverable errors have paths to send their error signals to the system configuration controller.

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

### BP23. Executing The WAKEUP Leaf of The GETSEC Instruction Multiple Times May Lead to a Machine Check Error

Problem: The GETSEC WAKEUP leaf broadcasts a wakeup message to all logical processors currently in a SENTER sleep state. It is sufficient to execute this instruction leaf once, per MLE (Measured Launch Event) launch, by the ILP (Initiating Logical Processor). Executing the leaf multiple times may lead to buffer entry corruptions resulting in machine check errors.

Implication: MLE launch may hang when GETSEC WAKEUP leaf is executed multiple times during the same launch.

Workaround: MLE launch software can workaround this erratum by avoiding multiple GETSEC WAKEUP leaf instruction executions.

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

### BP24. CKE-Lo Feature Can Not be Disabled When Memory Controller Transactions are Active

Problem: If the CKE-Lo (Clock Enable de-asserted) feature is disabled when the memory controller transactions are active, then it may cause the system to hang.

Implication: Disabling the CKE-Lo feature when the memory controller transactions are active, may result in the transactions timing out causing the system to hang.

Workaround: Software is required to quiesce memory traffic (including patrol scrub) before disabling the CKE-Lo feature.

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

### BP25. Executing The Intel TXT GETSEC SENTER Instruction Leaf May Lead to a Machine Check Error

Problem: The GETSEC SENTER instruction leaf broadcasts a message in order to handshake/ rendezvous between different logical processors. The processor uses incorrect byte-enables when broadcasting this message to remote processor sockets. This may result in a Machine Check error on multisocket platforms.

Implication: Due to this erratum, Intel TXT AC Modules cannot be run on multisocket platforms.

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*.

### BP26. Task Switch to a TSS With an Inaccessible LDTR Descriptor May Cause Unexpected Faults

Problem: A task switch may load the LDTR (Local Descriptor Table Register) with an incorrect segment descriptor if the LDT (Local Descriptor Table) segment selector in the new TSS specifies an inaccessible location in the GDT (Global Descriptor Table).

Implication: Future accesses to the LDT may result in unpredictable system behavior.

Workaround: Operating system code should ensure that segment selectors used during task switches to the GDT specify offsets within the limit of the GDT and that the GDT is fully paged into memory.

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

### BP27. An Intel® QPI Link Layer Retry Quickly Followed by an Intel® QPI Physical Layer Reset May Cause an MCE

Problem: While an Intel® QPI link is processing a link level retry requested by a remote Intel® QPI agent (due to link CRC errors), if an Intel® QPI phy layer reset is triggered and aligns with a specific retry stage, a packet may get dropped and cause time out error with MCA error code, IA32_MCi_Status [15:0] encoded as a Bus and Interconnect Error with Timeout [bit 8] = 1, Cache Hierarchy Error, or Internal Timer error.

Implication: Due to this erratum, a fatal MCE may be signaled with MCA error code, IA32_MCi_Status [15:0] encoded as a Bus and Interconnect Error with Timeout [bit 8] = 1, Cache Hierarchy Error, or Internal Timer error.

Workaround: None identified.

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

### BP28. LLC Arrays May have Incorrect Values after Warm Reset when Memory BIST is Disabled

Problem: When Memory BIST is disabled in the platform, LLC (Last Level Cache) arrays do not get initialized properly when coming out of warm reset.

Implication: Due to this erratum, data may be left valid in the LLC array which subsequently may be used/consumed by the processor during BIOS execution 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*.

### BP29. VM Entries that Return from SMM May Incorrectly Write to the SMRR Protected Region

Problem: If the executive-VMCS pointer field in the VMCS does not contain the VMXON pointer and the "use TPR shadow" VM-execution control is 1 in the executive VMCS, a VM entry that returns from SMM may write to the virtual-APIC page. Due to this erratum, this write may occur even if the virtual-APIC page is in the region protected by the SMRR (system-management range register).

Implication: The writes to the virtual-APIC page may corrupt data in SMRAM.

Workaround: If software sets the "use TPR shadow" VM-execution control to 1, it should not VMWRITE the virtual-APIC address to an address in the range protected by the SMRR.

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

### BP30. System Quiesce Events Initiated While Power Events are In Progress May Cause System Hangs

Problem: BIOS initiation of a system quiesce flow via the QUIESCE_CONTROL2 MSR (51H) and exit of a system quiesce flow QUIESCE_CONTROL1 MSR (50H) via may conflict with a power event on the BSP (Boot Strap Processor) core. Due to this conflict, the BSP core,

which BIOS code runs on, may have one thread take the power event and the other thread not take the power event, resulting in a system hang.

Implication: As a result of this erratum, the system may hang after BIOS initiates a system quiesce flow.

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*.

### BP31. Uncorrected Memory Error Detected by a Memory Patrol Scrub With SMI Generated by Other Memory Controllers May Cause MCE/System Management Interrupt Race Condition

Problem: BIOS may configure a System Management Interrupt to be signaled when the patrol scrub engine has reached the end of scrubbing a memory range. If the System Management Interrupt is generated while an uncorrected error is detected by another memory patrol scrub engine, it may result MCE/SMI race condition which may lead to system shutdown.

Implication: Due to this erratum, the system may shut down.

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*.

### BP32. Broken trace to either the P or the N lane of the Intel® SMI forwarded clock differential pair may result in loss of forwarded clock but not always lead to clock lane failover.

Problem: If either only the P or the N lane of the Intel® SMI forwarded clock is broken, then processor is capable of detecting minimum differential swing on the clock lane, thus resulting in the processor to assume that the forwarded clock still exists. Consequently, the processor will proceed to the Intel® SMI link training phase.

Implication: If the processor proceeds to the link training phase, then based on observations, it is possible that the Intel® SMI link may fail to train even after seven retry attempts and continue to remain in RESET state; or, if the link successfully reached L0 state, then the link may be unstable and shortly return to Disable_a state. However, if the P and N lanes of the forwarded clock differential pair are both broken due to board trace issues, then the clock failover mechanism on Intel® SMI channel has been found to operate successfully as expected.

Workaround: None identified.

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

### BP33. Package C3/C6 with Memory Self-refresh Enabled May Cause False Error Logging

Problem: When the processor is in Package C3/C6 with Memory Self Refresh, correctable errors may occur resulting in a system management interrupt (SMI). The SMI generation may result in a false error being logged in the IA32_MC6_STATUS (MSR 0x419) register.

Implication: Due to this erratum, a false error may be reported in IA32_MC6_STATUS register.

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*.

### BP34. Performance Monitor WOKEN Event May Under Count

Problem: Performance Monitoring counter WOKEN (Event: 0x0F8) counts the number of cores woken up from core C-states. Due to this erratum, the WOKEN event may not count the cores that are woken up from core C-states due to Trusted Execution Technology transactions.

Implication: Performance Monitoring Event WOKEN will under count the number of cores woken up from core C-states due to Trusted Execution Technology transaction.

Workaround: None identified.

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

### BP35. PECI Command Average Temperature Read does not report correct Temperature

Problem: The PECI (Platform Environment Control Interface) mailbox command 0x21, Average Temperature Read, is a feature which calculates the average temperature of the processor cores and reports it. In some instances the temperature reported out from the PECI Command Average Temperature Read is significantly higher than the actual processor average temperature.

Implication: The PECI Command Average Temperature Read may report a temperature that is higher than the actual processor average temperature.

Workaround: None identified.

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

### BP36. Intel® QPI Initialization May Cause a CATERR During Power-on Reset

Problem: In a complex set of circumstances during a power cycle reset, a CATERR may occur during the Intel® QPI initialization sequence as a result of a race condition where an Intel QPI completion transaction arrives while the receiver is still going through its initialization.

Implication: One of the application processors PBSP (package BSP) may cause a CATERR assertion resulting in a failure to complete BIOS post. This is only a boot issue and cannot occur during run-time.

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*.

### BP37. EOI Transaction May Not be Sent if Software Enters Core C6 During an Interrupt Service Routine

Problem: If core C6 is entered after the start of an interrupt service routine but before a write to the APIC EOI (End of Interrupt) register, and the core is woken up by an event other than a fixed interrupt source the core may drop the EOI transaction the next time APIC EOI register is written and further interrupts from the same or lower priority level will be blocked.

Implication: EOI transactions may be lost and interrupts may be blocked when core C6 is used during interrupt service routines.

Workaround: Software should check the ISR register and if any interrupts are in service only enter C1.

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

### BP38. 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*.

### BP39. 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 (that is, 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 (that is, 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 3A of the 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*.

### BP40. A Page Fault May Not be Generated When the PS bit is set to "1" in a PML4E or PDPTE

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

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

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

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

### BP41. IO_SMI Indication in SMRAM State Save Area May be Set Incorrectly

Problem: The IO_SMI bit in SMRAM's location 7FA4H is set to "1" by the processor to indicate a System Management Interrupt (SMI) occurred as the result of executing an instruction that reads from an I/O port. Due to this erratum, the IO_SMI bit may be incorrectly set by:

- A non-I/O instruction
- An event where an I/O read sets the IO_SMI bit but another interrupt is taken before the recognition of the SMI event
- A REP INS instruction
- An I/O read that redirects to MWAIT

Implication: SMM handlers may get false IO_SMI indication.

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*.

### BP42. Writing an Illegal Vector to the IA32_X2APIC_SELF_IPI MSR Will Hang the Processor

Problem: Writing an illegal vector (0 to 15) to the IA32_X2APIC_SELF_IPI MSR while the local APIC is in x2APIC mode will cause the processor to hang.

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

Workaround: Software should not write an illegal vector to the IA32_X2APIC_SELF_IPI MSR while the local APIC is in X2APIC mode. Virtual-machine monitors should not allow guest software to write to the IA32_X2APIC_SELF_IPI MSR.

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

### BP43. Successive Fixed Counter Overflows May be Discarded

Problem: Under specific internal conditions, when using Freeze PerfMon on PMI feature (bit 12 in IA32_DEBUGCTL.Freeze_PerfMon_on_PMI, MSR 1D9H), if two or more PerfMon Fixed Counters overflow very closely to each other, the overflow may be mishandled for some of them. This means that the counter's overflow status bit (in MSR_PERF_GLOBAL_STATUS, MSR 38EH) may not be updated properly; additionally, PMI interrupt may be missed if software programs a counter in Sampling-Mode (PMI bit is set on counter configuration).

Implication: Successive Fixed Counter overflows may be discarded when Freeze PerfMon on PMI is used.

Workaround: Software can avoid this by:

1. Avoid using Freeze PerfMon on PMI bit
2. Enable only one fixed counter at a time when using Freeze PerfMon on PMI

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

### BP44. VM Exits Due to "NMI-Window Exiting" May Not Occur Following a VM Entry to the Shutdown State

Problem: If VM entry is made with the "virtual NMIs" and "NMI-window exiting", VM-execution controls set to 1, and if there is no virtual-NMI blocking after VM entry, a VM exit with exit reason "NMI window" should occur immediately after VM entry unless the VM entry put the logical processor in the wait-for SIPI state. Due to this erratum, such VM exits do not occur if the VM entry put the processor in the shutdown state.

Implication: A VMM may fail to deliver a virtual NMI to a virtual machine in the shutdown state.

Workaround: Before performing a VM entry to the shutdown state, software should check whether the "virtual NMIs" and "NMI-window exiting" VM-execution controls are both 1. If they are, software should clear "NMI-window exiting" and inject an NMI as part of VM entry.

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

### BP45. Execution of INVVPID Outside 64-Bit Mode Cannot Invalidate Translations For 64-Bit Linear Addresses

Problem: Executions of the INVVPID instruction outside 64-bit mode with the INVVPID type "individual-address invalidation" ignore bits 63:32 of the linear address in the INVVPID descriptor and invalidate translations for bits 31:0 of the linear address.

Implication: The INVVPID instruction may fail to invalidate translations for linear addresses that set bits in the range 63:32. Because this erratum applies only to executions outside 64-bit mode, it applies only to attempts by a 32-bit virtual-machine monitor (VMM) to invalidate translations for a 64-bit guest. 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*.

### BP46. A Combination of Data Accesses That Are Split Across Cacheline Boundaries May Lead to a Processor Hang

Problem: Under certain complex micro-architectural conditions, closely spaced data accesses that are split across cacheline boundaries may lead to a processor hang.

Implication: Due to this erratum, the processor may hang. This erratum has not been observed with any general purpose operating systems.

Workaround: None identified.

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

### BP47. A Load May Appear to be Ordered Before an Earlier Locked Instruction

Problem: Under certain timing conditions involving multiple cores, a cacheable load may appear to be ordered before an earlier cacheable locked instruction that accesses a different location.

Implication: Locked instructions and subsequent loads may not occur in the expected order when run on multiple cores. In some circumstances this could lead to unpredictable system behavior.

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

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

### BP48. VMRESUME May Omit Check of Revision Identifier of Linked VMCS

Problem: If the VMCS link pointer is valid in the VMCS, VM entry instructions should check that the 32 bits referenced by that pointer contains the processor's VMCS revision identifier and fail if it does not. Due to this erratum, VMRESUME may omit this check and thus not cause VM entry to fail in some cases.

Implication: The revision identifier of the linked VMCS may not be checked. 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*.

### BP49. APIC Timer Interrupts May be Lost During Core C3

Problem: APIC timer interrupts intended to awaken from core C3 may be lost under certain timing conditions.

Implication: Due to this erratum, a lost timer interrupt may cause the system to 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*.

### BP50. 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 steppings affected, see the *Summary Tables of Changes*.

### BP51. CR0.CD Is Ignored in VMX Operation

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

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

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

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

### BP52. 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*.

### BP53. The Corrected Error Count Overflow Bit in IA32_ MC0_STATUS is Not Updated When The UC Bit is Set

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

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

Workaround: None identified.

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

### BP54. The Upper 32 Bits of CR3 May be Incorrectly Used With 32-Bit Paging

Problem: When 32-bit paging is in use, the processor should use a page directory located at the 32-bit physical address specified in bits 31:12 of CR3; the upper 32 bits of CR3 should be ignored. Due to this erratum, the processor will use a page directory located at the 64-bit physical address specified in bits 63:12 of CR3.

Implication: The processor may use an unexpected page directory or, if EPT (Extended Page Tables) is in use, cause an unexpected EPT violation. This erratum applies only if software enters 64-bit mode, loads CR3 with a 64-bit value, and then returns to 32-bit paging without changing CR3. Intel has not observed this erratum with any commercially available software.

Workaround: Software that has executed in 64-bit mode should reload CR3 with a 32-bit value before returning to 32-bit paging.

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

### BP55. EPT Violations May Report Bits 11:0 of Guest Linear Address Incorrectly

Problem: If a memory access to a linear address requires the processor to update an accessed or dirty flag in a paging-structure entry and if that update causes an EPT violation, the processor should store the linear address into the "guest linear address" field in the VMCS. Due to this erratum, the processor may store an incorrect value into bits 11:0 of this field. (The processor correctly stores the guest-physical address of the paging-structure entry into the "guest-physical address" field in the VMCS.)

Implication: Software may not be easily able to determine the page offset of the original memory access that caused the EPT violation. Intel has not observed this erratum to impact the operation of any commercially available software.

Workaround: Software requiring the page offset of the original memory access address can derive it by simulating the effective address computation of the instruction that caused the EPT violation.

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

### BP56. IA32_VMX_VMCS_ENUM MSR (48AH) Does Not Properly Report The Highest Index Value Used For VMCS Encoding

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

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

Workaround: None identified.

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

### BP57. Virtual-APIC Page Accesses With 32-Bit PAE Paging May Cause a System Crash

Problem: If a logical processor has EPT (Extended Page Tables) enabled, is using 32-bit PAE paging, and accesses the virtual-APIC page then a complex sequence of internal processor micro-architectural events may cause an incorrect address translation or machine check on either logical processor.

Implication: This erratum may result in unexpected faults, an uncorrectable TLB error logged in IA32_MCi_STATUS.MCACOD (bits [15:0]) with a value of 0000_0000_0001_xxxxb (where x stands for 0 or 1), a guest or hypervisor crash, or other 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*.

### BP58. VM Exit May Set IA32_EFER.NXE When IA32_MISC_ENABLE Bit 34 is Set to 1

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

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

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

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

### BP59. Performance Monitor Counter MEM_INST_RETIRED.STORES May Count Higher than Expected

Problem: Performance Monitoring counter MEM_INST_RETIRED.STORES (Event: 0BH, Umask: 02H) 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 MEM_INST_RETIRED.STORES may report counts higher than expected.

Workaround: None identified.

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

### BP60. 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*.

### BP61. An IRET Instruction That Results in a Task Switch Does Not Serialize The Processor

Problem: An IRET instruction that results in a task switch by returning from a nested task does not serialize the processor (contrary to the Software Developer's Manual Vol. 3 section titled "Serializing Instructions").

Implication: Software which depends on the serialization property of IRET during task switching may not behave as expected. Intel has not observed this erratum to impact the operation of any commercially available software.

Workaround: None identified. Software can execute an MFENCE instruction immediately prior to the IRET instruction if serialization is needed.

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

# Intel® Xeon® Processor E7-8800/4800/2800 Product Families BIOS ACM Errata

## AC1. BIOS ACM Exit INIT (LockConfig) Call May Fail on Certain IOH Bus Configurations

Problem: With certain IOH bus configurations, the BIOS ACM Exit Init (LockConfig) call may be unable to lock the IOHs and the call will fail.

Implication: When this erratum occurs, the TXT.HEAP.BASE and TXT.HEAP.SIZE registers will be locked, and BIOS will be unable to setup TXT heap memory for MLE (Measured Launch Environment) boot.

Workaround: On single IOH configurations, the IOH can use bus 0x00-0x7F to avoid this erratum. On dual IOH configurations, IOH0 can use bus 00-0x7F and IOH1 can use bus 0x80-0xF7 to avoid this erratum.

Status: Fixed in BIOS ACM 1.1.

# Intel® Xeon® Processor E7-8800/4800/2800 Product Families SINIT ACM Errata

### AS1. TXT.ERRORCODE TPM Command Return Code And Launch Control Policy List Index And Minor Code Are Not Reported Correctly.

Problem: On affected SINIT ACM releases, the TXT.ERRORCODE register TPM command return code (bits 24:16), Launch Control Policy List Index (bits 24:22) and Launch Control Policy Minor Code (bits 21:16) are not reported correctly.

Implication: Software depending upon TXT.ERRORCODE error reporting for the TPM command return code, Launch Control Policy List Index, or Launch Control Policy Minor Code may not behave as expected.

Workaround: None.

Status: See Intel® Xeon® Processor E7-8800/4800/2800 Product Families SINIT ACM Errata Summary for affected releases.

### AS2. SINIT Buffer Overflow Vulnerability

Problem: SINIT Authenticated Code Module (ACM) 1.0 is susceptible to a buffer overflow issue.

Implication: When Intel® Trusted Execution Technology measured launch is invoked using SINIT Authenticated Code Module 1.0, the platform is susceptible to an OS kernel-level exploit which may compromise certain SINIT ACM functionality.

Workaround: It is possible for a BIOS update and an updated SINIT ACM 1.1 to be used as a workaround for this erratum. Previous SINIT ACM releases will no longer function with the BIOS update.

Status: See Intel® Xeon® Processor E7-8800/4800/2800 Product Families SINIT ACM Errata Summary for affected releases.

# Specification Changes

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

- Intel® Xeon® Processor E7-8800/4800/2800 Product Families Datasheet, Volumes 1 and 2
- Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1: Basic Architecture
- Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A: Instruction Set Reference Manual A-M
- Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2B: Instruction Set Reference Manual N-Z
- Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3A: System Programming Guide
- Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3B: System Programming Guide

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

## SCh1. Correction to Product Families Features

In the *Intel® Xeon® Processor E7-8800/4800/2800 Product Families Datasheet, Volume 2* under section 1.1, stated:

- Support for 1.5 V/1.35 V High Density Reduced Load RDIMMs (also called LRDIMM, which is Load Reduced DIMM)

is changing to:

- Support for LRDIMM removed

# Specification Clarifications

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

- *Intel® Xeon® Processor E7-8800/4800/2800 Product Families Datasheet, Volumes 1 and 2*.
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 1: Basic Architecture*.
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2A: Instruction Set Reference Manual A-M*.
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 2B: Instruction Set Reference Manual N-Z*.
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3A: System Programming Guide*.
- *Intel® 64 and IA-32 Architectures Software Developer's Manual, Volume 3B: System Programming Guide*.

## SC1. Package C3/C6 Memory Self-Refresh Error Handling

The following sub-bullet will be added to section 9.5.1.2 of the Intel® Xeon® Processor E7-8800/4800/2800 Product Families Datasheet:

- Frequent transitions to package C3/C6 memory self-refresh during some idle workloads may increase the number of recoverable (corrected) SMI link events observed. While a low occurrence of recoverable events is normal for high speed processor busses, the rate of recoverable events may increase during some idle conditions due to the higher number of SMI link disable-enable transitions occurring.

# Documentation Changes

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

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

All Documentation Changes will be incorporated into a future version of the appropriate Processor 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/products/processor/manuals/index.htm

## DC1. 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 Intel® 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 with any of the following CPUID DisplayFamily_DisplayModel signatures [listed in Table 8]. 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 7. CPUID DisplayFamily_DisplayModel Signatures for Legacy Processors That Resolve to Higher Performance Setting of Conflicting Duty Cycle Requests**

| | | | | | | |
|---|---|---|---|---|---|---|
| 06_1A | 06_1E | 06_25 | 06_27 | 06_2E | 06_35 | 0F_xx |
| 06_1C | 06_1F | 06_26 | 06_2C | 06_2F | 06_36 | |

§