



Revision Guide for AMD Family 15h Models 00h-0Fh Processors

Publication # 48063 Revision: 3.04 Issue Date: November 2011
--

Advanced Micro Devices 

© 2010-2011 Advanced Micro Devices, Inc. All rights reserved.

The contents of this document are provided in connection with Advanced Micro Devices, Inc. (“AMD”) products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. The information contained herein may be of a preliminary or advance nature and is subject to change without notice. No license, whether express, implied, arising by estoppel or otherwise, to any intellectual property rights is granted by this publication. Except as set forth in AMD’s Standard Terms and Conditions of Sale, AMD assumes no liability whatsoever, and disclaims any express or implied warranty, relating to its products including, but not limited to, the implied warranty of merchantability, fitness for a particular purpose, or infringement of any intellectual property right.

AMD’s products are not designed, intended, authorized or warranted for use as components in systems intended for surgical implant into the body, or in other applications intended to support or sustain life, or in any other application in which the failure of AMD’s product could create a situation where personal injury, death, or severe property or environmental damage may occur. AMD reserves the right to discontinue or make changes to its products at any time without notice.

Trademarks

AMD, the AMD Arrow logo, AMD FX, AMD Opteron, and combinations thereof, are trademarks of Advanced Micro Devices, Inc.

HyperTransport is a licensed trademark of the HyperTransport Technology Consortium.

Other product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

List of Tables

Table 1. Arithmetic and Logic Operators	8
Table 2. CUID Values for AMD Family 15h Models 00h-0Fh G34r1 Processor Revisions	9
Table 3. CUID Values for AMD Family 15h Models 00h-0Fh C32r1 Processor Revisions	9
Table 4. CUID Values for AMD Family 15h Models 00h-0Fh AM3r2 Processor Revisions	10
Table 5. Cross Reference of Product Revision to OSVW ID	12
Table 6. Cross-Reference of Processor Revision to Errata	13
Table 7. Cross-Reference of Errata to Package Type	15
Table 8. Cross-Reference of Errata to Processor Segments	17

List of Figures

Figure 1. Format of CPUID Fn0000_0001_EAX9

Revision History

Date	Revision	Description
November 2011	3.04	Added errata #600 and #693 - #695 . Updated errata #534 and #671 . Added package types G34, C32 to Table 7 . Added AMD Opteron™ 4200 Series Processor and AMD Opteron™ 6200 Series Processor to Overview and Table 8 . Added Table 2 and Table 3 .
October 2011	3.00	Initial public release.

Overview

The purpose of the *Revision Guide for AMD Family 15h Models 00h-0Fh Processors* is to communicate updated product information to designers of computer systems and software developers. This revision guide includes information on the following products:

- AMD FX™-Series Processor
- AMD Opteron™ 4200 Series Processor
- AMD Opteron™ 6200 Series Processor

This guide consists of these major sections:

- [Processor Identification](#) shows how to determine the processor revision and workaround requirements, and to construct, program, and display the processor name string.
- [Product Errata](#) provides a detailed description of product errata, including potential effects on system operation and suggested workarounds. An erratum is defined as a deviation from the product's specification, and as such may cause the behavior of the processor to deviate from the published specifications.
- [Documentation Support](#) provides a listing of available technical support resources.

Revision Guide Policy

Occasionally, AMD identifies product errata that cause the processor to deviate from published specifications. Descriptions of identified product errata are designed to assist system and software designers in using the processors described in this revision guide. This revision guide may be updated periodically.

Conventions

Numbering

- **Binary numbers.** Binary numbers are indicated by appending a "b" at the end, e.g., 0110b.
- **Decimal numbers.** Unless specified otherwise, all numbers are decimal. This rule does not apply to the register mnemonics.
- **Hexadecimal numbers.** Hexadecimal numbers are indicated by appending an "h" to the end, e.g., 45F8h.
- **Underscores in numbers.** Underscores are used to break up numbers to make them more readable. They do not imply any operation. e.g., 0110_1100b.
- **Undefined digit.** An undefined digit, in any radix, is notated as a lower case "x".

Register References and Mnemonics

In order to define errata workarounds it is sometimes necessary to reference processor registers. References to registers in this document use a mnemonic notation consistent with that defined in the *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 00h-0Fh Processors*, order# 42301. Each mnemonic is a concatenation of the register-space indicator and the offset of the register. The mnemonics for the various register spaces are as follows:

- **IOXXX:** x86-defined input and output address space registers; XXX specifies the byte address of the I/O register in hex (this may be 2 or 3 digits). This space includes the I/O-Space Configuration Address Register (IOCF8) and the I/O-Space Configuration Data Port (IOCFC) to access configuration registers.
- **DZFYxXXX:** PCI-defined configuration space at bus 0; Z specifies the PCI device address in hex; XXX specifies the byte address of the configuration register (this may be 2 or 3 digits) in hex; Y specifies the function number. For example, D18F3x40 specifies the register at bus 0, device 18h, function 3, address 40h. Some registers in D18F2xXXX have a `_dct[1:0]` mnemonic suffix, which indicates there is one instance per DRAM controller (DCT). The DCT instance is selected by DCT Configuration Select[DctCfgSel] (D18F1x10C[0]).
- **DZFYxXXX_xZZZZZ:** Port access through the PCI-defined configuration space at bus 0; Z specifies the PCI device address in hex; XXX specifies the byte address of the data port configuration register (this may be 2 or 3 digits) in hex; Y specifies the function number; ZZZZZ specifies the port address (this may be 2 to 7 digits) in hex. For example, D18F2x9C_x1C specifies the port 1Ch register accessed using the data port register at bus 0, device 18h, function 2, address 9Ch. Refer to the *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 00h-0Fh Processors*, order# 42301 for access properties. Some registers in D18F2xXXX_xZZZZZ have a `_dct[1:0]` mnemonic suffix, which indicates there is one instance per DRAM controller (DCT). The DCT instance is selected by DCT Configuration Select[DctCfgSel] (D18F1x10C[0]).
- **APICXXX:** APIC memory-mapped registers; XXX is the byte address offset from the base address in hex (this may be 2 or 3 digits). The base address for this space is specified by the APIC Base Address Register (APIC_BAR) at MSR0000_001B.
- **CPUID FnXXXX_XXXX_RRR_xYYY:** processor capability information returned by the CPUID instruction where the CPUID function is XXXX_XXXX (in hex) and the ECX input is YYY (if specified). When a register is specified by RRR, the reference is to the data returned in that register. For example, CPUID Fn8000_0001_EAX refers to the data in the EAX register after executing CPUID instruction function 8000_0001h.
- **MSRXXXX_XXXX:** model specific registers; XXXX_XXXX is the MSR number in hex. This space is accessed through x86-defined RDMSR and WRMSR instructions.

- **PMCxXXX[Y]**: performance monitor events; XXX is the hexadecimal event counter number programmed into MSRC001_020[A,8,6,4,2,0][EventSelect] (PERF_CTL[5:0] bits 7:0). Y, when specified, signifies the unit mask programmed into MSRC001_020[A,8,6,4,2,0][UnitMask] (PERF_CTL[5:0] bits 15:8).
- **NBPMCxXXX[Y]**: northbridge performance monitor events; XXX is the hexadecimal event counter number programmed into MSRC001_024[6,4,2,0][EventSelect] (NB_PERF_CTL[3:0] bits 7:0). Y, when specified, signifies the unit mask programmed into MSRC001_024[6,4,2,0][UnitMask] (NB_PERF_CTL[3:0] bits 15:8).

Many register references use the notation "[]" to identify a range of registers. For example, D18F2x[1,0][4C:40] is a shorthand notation for D18F2x40, D18F2x44, D18F2x48, D18F2x4C, D18F2x140, D18F2x144, D18F2x148, and D18F2x14C.

Arithmetic and Logical Operators

In this document, formulas follow some Verilog conventions as shown in [Table 1](#) .

Table 1. Arithmetic and Logic Operators

Operator	Definition
{ }	Curly brackets are used to indicate a group of bits that are concatenated together. Each set of bits is separated by a comma. E.g., {Addr[3:2], Xlate[3:0]} represents a 6-bit value; the two MSBs are Addr[3:2] and the four LSBs are Xlate[3:0].
	Bitwise OR operator. E.g. (01b 10b == 11b).
	Logical OR operator. E.g. (01b 10b == 1b); logical treats multibit operand as 1 if >=1 and produces a 1-bit result.
&	Bitwise AND operator. E.g. (01b & 10b == 00b).
&&	Logical AND operator. E.g. (01b && 10b == 1b); logical treats multibit operand as 1 if >=1 and produces a 1-bit result.
^	Bitwise exclusive-OR operator; sometimes used as "œraised to the power of" as well, as indicated by the context in which it is used. E.g. (01b ^ 10b == 11b). E.g. (2^2 == 4).
~	Bitwise NOT operator (also known as one's complement). E.g. (~10b == 01b).
!	Logical NOT operator. E.g. (!10b == 0b); logical treats multibit operand as 1 if >=1 and produces a 1-bit result.
==	Logical "is equal to" operator.
!=	Logical "is not equal to" operator.
<=	Less than or equal operator.
>=	Greater than or equal operator.
*	Arithmetic multiplication operator.
/	Arithmetic division operator.
<<	Shift left first operand by the number of bits specified by the 2nd operand. E.g. (01b << 01b == 10b).
>>	Shift right first operand by the number of bits specified by the 2nd operand. E.g. (10b >> 01b == 01b).

Processor Identification

This section shows how to determine the processor revision, program and display the processor name string, and construct the processor name string.

Revision Determination

A processor revision is identified using a unique value that is returned in the EAX register after executing the CPUID instruction function 0000_0001h (CPUID Fn0000_0001_EAX). [Figure 1](#) shows the format of the value from CPUID Fn0000_0001_EAX. In some cases, two or more processor revisions may exist within a stepping of a processor family and are identified by a unique value in D18 F4x164 Fixed Errata Register (see [D18F4x164 Fixed Errata Register](#)).

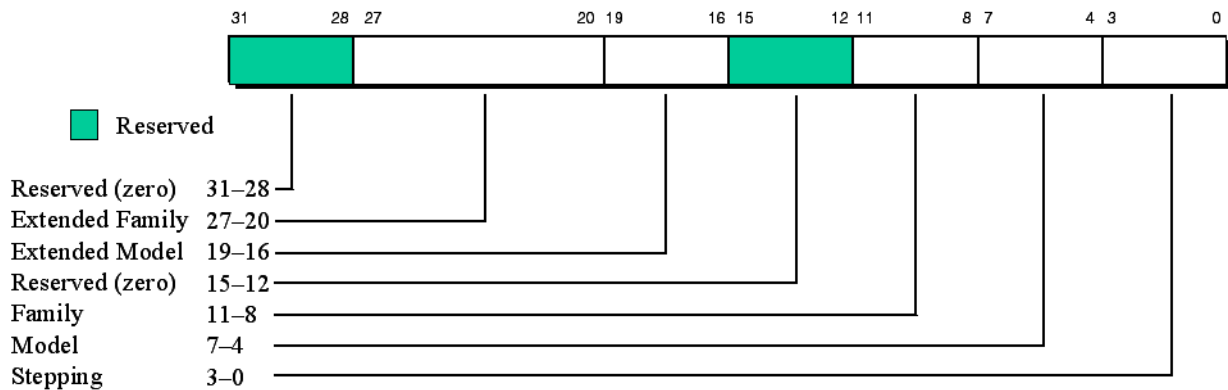


Figure 1. Format of CPUID Fn0000_0001_EAX

[Table 2](#) through [Table 4](#) cross-references the identification number from CPUID Fn0000_0001_EAX and D18F4x164 (if necessary) for each revision of the processor to each processor segment. "X" signifies that the revision has been used in the processor segment. "N/A" signifies that the revision has not been used in the processor segment.

Table 2. CPUID Values for AMD Family 15h Models 00h-0Fh G34r1 Processor Revisions

CPUID Fn0000_0001_EAX, D18F4x164[0] (Mnemonic)	AMD Opteron™ 6200 Series Processor
00600F12h, 1b (OR-B2)	X

Table 3. CPUID Values for AMD Family 15h Models 00h-0Fh C32r1 Processor Revisions

CPUID Fn0000_0001_EAX, D18F4x164[0] (Mnemonic)	AMD Opteron™ 4200 Series Processor
00600F12h, 1b (OR-B2)	X

Table 4. CPUID Values for AMD Family 15h Models 00h-0Fh AM3r2 Processor Revisions

CPUID Fn0000_0001_EAX, D18F4x164[0]	AMD FX™-Series Processor
00600F12h, 1b (OR-B2)	X

D18F4x164 Fixed Errata Register

Communicating the status of an erratum within a stepping of a processor family is necessary in certain circumstances. D18F4x164 is used to communicate the status of such an erratum fix so that BIOS or system software can determine the necessity of applying the workaround. Under these circumstances, the erratum workaround references the specified bit to enable software to test for the presence of the erratum. The erratum may be specific to some steppings of the processor, and the specified bit may or may not be set on other unaffected revisions within the same family. Therefore, software should use the CPUID Fn00000_0001_EAX extended model, model, and stepping as the first criteria to identify the applicability of an erratum. Once defined, the definition of the status bit will persist within the family of processors.

Bits	Description
31:0	0000_0001h. Reserved.

Programming and Displaying the Processor Name String

This section, intended for BIOS programmers, describes how to program and display the 48-character processor name string that is returned by CPUID Fn8000_000[4:2]. The hardware or cold reset value of the processor name string is 48 ASCII NUL characters, so the BIOS must program the processor name string before any general purpose application or operating system software uses the extended functions that read the name string. It is common practice for the BIOS to display the processor name string and model number whenever it displays processor information during boot up.

Note: Motherboards that do not program the proper processor name string and model number will not pass AMD validation and will not be posted on the AMD Recommended Motherboard Web site.

The name string must be ASCII NUL terminated and the 48-character maximum includes that NUL character.

The processor name string is programmed by MSR writes to the six MSR addresses covered by the range MSRC001_00[35:30]h. Refer to the *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h*

Models 00h-0Fh Processors, order# 42301, for the format of how the 48-character processor name string maps to the 48 bytes contained in the six 64-bit registers of MSRC001_00[35:30].

The processor name string is read by CPUID reads to a range of CPUID functions covered by CPUID Fn8000_000[4:2]. Refer to CPUID Fn8000_000[4:2] in the *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 00h-0Fh Processors*, order# 42301, for the 48-character processor name string mapping to the 48 bytes contained in the twelve 32-bit registers of CPUID Fn8000_000[4:2].

Constructing the Processor Name String

This section describes how to construct the processor name string. BIOS uses the following fields to create the name string:

The name string is formed as follows:

1. If D18F5x198_x0 is 00000000h, then use a name string of "AMD Unprogrammed Engineering Sample" and skip the remaining steps.
2. Read {D18F5x198_x1, D18F5x198_x0} and write this value to MSRC001_0030.
3. Read {D18F5x198_x3, D18F5x198_x2} and write this value to MSRC001_0031.
4. Read {D18F5x198_x5, D18F5x198_x4} and write this value to MSRC001_0032.
5. Read {D18F5x198_x7, D18F5x198_x6} and write this value to MSRC001_0033.
6. Read {D18F5x198_x9, D18F5x198_x8} and write this value to MSRC001_0034.
7. Read {D18F5x198_xB, D18F5x198_xA} and write this value to MSRC001_0035.

Operating System Visible Workarounds

This section describes how to identify operating system visible workarounds.

MSRC001_0140 OS Visible Work-around MSR0 (OSVW_ID_Length)

This register, as defined in *AMD64 Architecture Programmer's Manual Volume 2: System Programming*, order# 24593, is used to specify the number of valid status bits within the OS Visible Work-around status registers.

The reset default value of this register is 0000_0000_0000_0000h.

BIOS shall program the OSVW_ID_Length to 0004h prior to hand-off to the OS.

Bits	Description
63:16	Reserved.
15:0	OSVW_ID_Length : OS visible work-around ID length. Read-write.

MSRC001_0141 OS Visible Work-around MSR1 (OSVW_Status)

This register, as defined in *AMD64 Architecture Programmer's Manual Volume 2: System Programming*, order# 24593, provides the status of the known OS visible errata. Known errata are assigned an OSVW_ID corresponding to the bit position within the valid status field.

Operating system software should use MSRC001_0140 to determine the valid length of the bit status field. For all valid status bits: 1=Hardware contains the erratum, and an OS software work-around is required or may be applied instead of a BIOS workaround. 0=Hardware has corrected the erratum, so an OS software work-around is not necessary.

The reset default value of this register is 0000_0000_0000_0000h.

Bits	Description
63:4	OsvwStatusBits : Reserved. OS visible work-around status bits. Read-write.
3	OsvwId3 : Reserved, must be zero.
2	OsvwId2 : Reserved, must be zero.
1	OsvwId1 : Reserved, must be zero.
0	OsvwId0 : Reserved, must be zero.

BIOS shall program the state of the valid status bits as shown in [Table 5](#) prior to hand-off to the OS.

Table 5. Cross Reference of Product Revision to OSVW ID

CPUID Fn0000_0001_EAX (Mnemonic)	MSRC001_0141 Bits
00600F12h (OR-B2)	0000_0000_0000_0000h

Product Errata

This section documents product errata for the processors. A unique tracking number for each erratum has been assigned within this document for user convenience in tracking the errata within specific revision levels. This table cross-references the revisions of the part to each erratum. An "X" indicates that the erratum applies to the revision. The absence of an "X" indicates that the erratum does not apply to the revision. An "*" indicates advance information that the erratum has been fixed but not yet verified. "No fix planned" indicates that no fix is planned for current or future revisions of the processor.

Note: There may be missing errata numbers. Errata that do not affect this product family do not appear. In addition, errata that have been resolved from early revisions of the processor have been deleted, and errata that have been reconsidered may have been deleted or renumbered.

Table 6. Cross-Reference of Processor Revision to Errata

No.	Errata Description	CPUID Fn0000_0001_EAX, D18F4x164[0]
		00600F12h 1b (OR-B2)
361	Breakpoint Due to An Instruction That Has an Interrupt Shadow May Be Lost	No fix planned
503	APIC Task-Priority Register May Be Incorrect	No fix planned
504	Corrected L3 Errors May Lead to System Hang	No fix planned
505	Scrub Rate Control Register Address Depends on DctCfgSel	No fix planned
520	Some Lightweight Profiling Counters Stop Counting When Instruction-Based Sampling is Enabled	No fix planned
534	Instruction-Based Sampling May Be Inaccurate	No fix planned
535	Lightweight Profiling May Not Indicate Fused Branch	No fix planned
536	Performance Counter for Instruction Cache Misses Does Not Increment for Sequential Prefetches	No fix planned
537	Performance Counter for Ineffective Software Prefetches Does Not Count for L2 Hits	No fix planned
538	Performance Counter Does Not Count for Some Retired Micro-Ops	No fix planned
540	GART Table Walk Probes May Cause System Hang	No fix planned
550	Latency Performance Counters Are Not Accurate	No fix planned
585	Incorrect Memory Controller Operation Due to a WrDatGrossDly Setting of 3.5 MEMCLKs	No fix planned
586	A Far Control Transfer Changing Processor Operating Mode May Generate a False Machine Check	No fix planned
592	VPEXTRQ and VPINSRQ May Not Signal Invalid-Opcode Exception	No fix planned
593	Last-Branch Record Enabled May Cause Machine Check and Incorrect LastBranchToIp	No fix planned
600	HyperTransport™ Link Retry Due to Partial CRC Error May Cause System Hang	No fix planned

No.	Errata Description	CPUID Fn0000_0001_EAX, D18F4x164[0]
		00600F12h 1b (OR-B2)
602	HyperTransport™ Link Frequency Changes May Cause a System Hang	No fix planned
608	P-state Limit Changes May Not Generate Interrupts	No fix planned
619	Non-Posted Reads May Block Write Dependent on Probe Responses	No fix planned
623	Small Code Segment Limits May Cause Incorrect Limit Faults	No fix planned
624	SB-RMI Processor State Accesses May Persistently Timeout if Interrupted by a Warm Reset	No fix planned
625	SB-RMI Writes May Not Be Observed by Processor	No fix planned
636	Instruction Addresses Near Canonical Address Limit May Cause #GP Exception	No fix planned
637	Processor Does Not Report the Correct DRAM Address for MCA Errors Within the CC6 Save Area	No fix planned
657	MC1_STATUS Enable Bit Not Set When Logging Corrected Errors	No fix planned
658	CPUID Incorrectly Reports Large Page Support in L2 Instruction TLB	No fix planned
659	VMCB Interrupt Shadow Status May Be Incorrect	No fix planned
660	APERF May Increase Unpredictably	No fix planned
661	P-State Limit and Stop Clock Assertion May Cause System Hang	No fix planned
663	Local Interrupts LINT0/LINT1 May Occur While APIC is Software Disabled	No fix planned
667	Processor May Generate Incorrect P-state Limit Interrupts	No fix planned
668	Load Operation May Receive Incorrect Data After Floating-point Exception	No fix planned
671	Debug Breakpoint on Misaligned Store May Cause System Hang	No fix planned
672	SVM Guest Performance Counters May Be Inaccurate Due to SMI	No fix planned
673	Misaligned Page Crossing String Operations May Cause System Hang	No fix planned
674	Processor May Cache Prefetched Data from Remapped Memory Region	No fix planned
675	Instructions Performing Read-Modify-Write May Alter Architectural State Before #PF	No fix planned
689	AM3r2 Six Core Processor May Limit P-state When Core C6 State is Disabled	No fix planned
690	Northbridge FIFO Read/Write Pointer Overlap May Cause Hang or Protocol Error Machine Check	No fix planned
693	Performance Counter May Incorrectly Count MXCSR Loads	No fix planned
694	IBS Sampling of Instruction Fetches May Be Uneven	No fix planned
695	Processor May Interpret FCW Incorrectly After FNSAVE/FSAVE Limit Fault	No fix planned

Cross-Reference of Errata to Package Type

This table cross-references the errata to each package type. "X" signifies that the erratum applies to the package type. An empty cell signifies that the erratum does not apply. An erratum may not apply to a package type due to a specific characteristic of the erratum, or it may be due to the affected silicon revision(s) not being used in this package.

Table 7. Cross-Reference of Errata to Package Type

Errata	Package		
	AM3r2	C32r1	G34r1
361	X	X	X
503	X	X	X
504	X	X	X
505	X	X	X
520	X	X	X
534	X	X	X
535	X	X	X
536	X	X	X
537	X	X	X
538	X	X	X
540	X	X	X
550	X	X	X
585	X	X	X
586	X	X	X
592	X	X	X
593	X	X	X
600		X	X
602	X	X	X
608	X	X	X
619	X	X	X
623	X	X	X
624	X	X	X
625	X	X	X
636	X	X	X
637	X	X	X
657	X	X	X
658	X	X	X
659	X	X	X

Errata	Package		
	AM3r2	C32r1	G34r1
660	X	X	X
661	X	X	X
663	X	X	X
667	X	X	X
668	X	X	X
671	X	X	X
672	X	X	X
673	X	X	X
674	X	X	X
675	X	X	X
689	X		
690	X	X	X
693	X	X	X
694	X	X	X
695	X	X	X

Cross-Reference of Errata to Processor Segments

This table cross-references the errata to each processor segment. "X" signifies that the erratum applies to the processor segment. An empty cell signifies that the erratum does not apply. An erratum may not apply to a processor segment due to a specific characteristic of the erratum, or it may be due to the affected silicon revision(s) not being used in this processor segment.

Table 8. Cross-Reference of Errata to Processor Segments

Errata	Processor Segment		
	AMD FX™ -Series Processor	AMD Opteron™ 4200 Processor	AMD Opteron™ 6200 Processor
361	X	X	X
503	X	X	X
504	X	X	X
505	X	X	X
520	X	X	X
534	X	X	X
535	X	X	X
536	X	X	X
537	X	X	X
538	X	X	X
540	X	X	X
550	X	X	X
585	X	X	X
586	X	X	X
592	X	X	X
593	X	X	X
600		X	X
602	X	X	X
608	X	X	X
619	X	X	X
623	X	X	X
624	X	X	X
625	X	X	X

Errata	Processor Segment		
	AMD FX™ -Series Processor	AMD Opteron™ 4200 Processor	AMD Opteron™ 6200 Processor
636	X	X	X
637	X	X	X
657	X	X	X
658	X	X	X
659	X	X	X
660	X	X	X
661	X	X	X
663	X	X	X
667	X	X	X
668	X	X	X
671	X	X	X
672	X	X	X
673	X	X	X
674	X	X	X
675	X	X	X
689	X		
690	X	X	X
693	X	X	X
694	X	X	X
695	X	X	X

361 Breakpoint Due to An Instruction That Has an Interrupt Shadow May Be Lost

Description

A #DB exception occurring in guest mode may be discarded under the following conditions:

- A trap-type #DB exception is generated in guest mode during execution of an instruction with an interrupt shadow, and
- The instruction that generated the exception is immediately followed by an instruction resulting in #VMEXIT.

Potential Effect on System

None expected under normal conditions. Debug exceptions may not be received for programs running under a hypervisor.

Suggested Workaround

None.

Fix Planned

No fix planned

503 APIC Task-Priority Register May Be Incorrect

Description

An APIC task priority register (TPR) write may use an incorrect internal buffer for the data.

Potential Effect on System

Incorrect interrupt prioritization.

Suggested Workaround

BIOS should set MSRC001_102A[11] to 1b.

Fix Planned

No fix planned

504 Corrected L3 Errors May Lead to System Hang

Description

Under a highly specific and detailed set of internal timing conditions that involves corrected L3 errors, a processor read from the L3 cache may hang.

Potential Effect on System

System hang.

Suggested Workaround

BIOS should program D18F3x1B8[18] to 1b.

Fix Planned

No fix planned

505 Scrub Rate Control Register Address Depends on DctCfgSel

Description

When DCT Configuration Select[DctCfgSel] (D18F1x10C[0]) is 1b, accesses to the Scrub Rate Control register (D18F3x58) incorrectly accesses a different register that does not actually affect any hardware.

Potential Effect on System

Incorrect scrub rate controls may be read or in effect.

Suggested Workaround

Software should clear DctCfgSel (D18F1x10C[0]) to 0b prior to any access to D18F3x58 Scrub Rate Control Register. The software must serialize any accesses to D18F3x58 with other accesses to registers that use DctCfgSel. When enabling scrub settings, BIOS should write D18F3x58 twice with the same value - once with D18F1x10C[0] set to 0b and once with D18F1x10C[0] set to 1b. BIOS should program D18F1x10C[0] to 0b before handing over control to the operating system .

Fix Planned

No fix planned

520 Some Lightweight Profiling Counters Stop Counting When Instruction-Based Sampling is Enabled

Description

When Lightweight Profiling (LWP) and Instruction-Based Sampling (IBS) measurement of instruction execution are simultaneously enabled, the following LWP counters do not increment:

- Instructions retired event counter (LWP EventId 2)
- Branches retired event counter (LWP EventId 3)

LWP is enabled once software executes a LLWCP or XRSTOR instruction with a valid LWPCB address. IBS instruction execution sampling is enabled when IBS Execution Control[IbsOpEn] (MSRC001_1033[17]) is 1b.

Potential Effect on System

Performance monitoring software using LWP may not have a count of instructions retired or branches retired.

Suggested Workaround

None.

Fix Planned

No fix planned

534 Instruction-Based Sampling May Be Inaccurate

Description

The processor may experience sampling inaccuracies when Instruction-Based Sampling (IBS) is enabled in the following cases:

- The processor may set IBS Op Data 3 Register[IbsDcStToLdCan, IbsDcStToLdFwd] (MSRC001_1037[12, 11]) incorrectly for load instructions that are tagged for IBS if there is a nearby store instruction that matches the load address in bits 11:0 .
- When IBS Op Data 3 Register[IbsDcMiss] (MSRC001_1037[7]) is 0b, IBS Op Data 3 Register[IbsDcMabHit] (MSRC001_1037[16]) should always be 0b. However, the processor may incorrectly set IbsDcMabHit .
- When the processor samples an instruction that is altering the CS_BASE value, the IbsOpRip reported in IBS Op Logical Address (MSRC001_1034) may be determined using an incorrect CS_BASE .
- In rare instances, a tagged branch may set an inaccurate value in IBS Branch Target Address Register (MSRC001_103B).
- When performing an IBS execution sample, the processor only sets, but never clears, the following bits:
 - IbsDcL2TlbMiss (MSR C001_1037[3])
 - IbsDcL2TlbHit2M (MSR C001_1037[6])
 - IbsDcL2TlbHit1G (MSR C001_1037[19])
- The processor incorrectly updates IBS Op Data 2 Register[NbIbsReqCacheHitSt, NbIbsReqDstProc, NbIbsReqSrc] (MSR C001_1036[5,4,2:0]) during an IBS fetch sample . If both IBS execution sampling (IBS Execution Control[IbsOpEn], MSRC001_1033[17] = 1b) and IBS fetch sampling (IBS Fetch Control[IbsFetchEn], MSRC001_1030[48] = 1b) are enabled simultaneously, valid execution sample data may be overwritten by a fetch sample resulting in IBS data that is inconsistent with the accompanying IBS execution sample data.

Potential Effect on System

Inaccuracies in performance monitoring software may be experienced.

Suggested Workaround

The following workarounds can be used for the above issues:

- No workaround exists for IbsDcStToLdCan and IbsDcStToLdFwd. These bits would not significantly over-indicate a store to load forwarding with most code.
- Performance monitoring software should treat IBS Op Data 3 Register[IbsDcMabHit] (MSRC001_1037[16]) as 0b when MSRC001_1037[7] is 0b.
- No workaround is necessary for IbsOpRip when CS_BASE is zero. CS_BASE is normally zero in commercially available software.
- Performance monitoring software should not rely on the value in MSRC001_103B (IbsBrTarget).
- Performance monitoring software should clear the IBS Op Data 3 Register (MSR C001_1037[63:0] = 0) between each IBS sample.
- Performance monitoring software should not rely on MSRC001_1036 if both IBS execution sampling and IBS fetch sampling are enabled simultaneously.

Fix Planned

No fix planned

535 Lightweight Profiling May Not Indicate Fused Branch

Description

The Lightweight Profiling (LWP) fused operation bit (FUS - bit 28 of the branch retired event record, LWP EventId 3) may not be set when the processor core is profiling a fused branch (a compare operation followed by a conditional branch that is executed as a single operation internally) and a #PF or nested-paging exception occurs during the storing of the event.

Potential Effect on System

Performance monitoring software may not profile a fused branch correctly.

Suggested Workaround

None.

Fix Planned

No fix planned

536 Performance Counter for Instruction Cache Misses Does Not Increment for Sequential Prefetches

Description

PMCx081 (Instruction Cache Misses) does not increment for L1 instruction cache misses that are due to sequential prefetches .

Potential Effect on System

Performance monitoring software may undercount instruction cache misses.

Suggested Workaround

Performance monitoring software may use the difference of PMCx083 and PMCx082 as a close approximation of instruction cache misses.

Fix Planned

No fix planned

537 Performance Counter for Ineffective Software Prefetches Does Not Count for L2 Hits

Description

PMCx052[3] (ineffective software prefetch due to an L2 cache hit) does not increment.

Potential Effect on System

Performance monitoring software can not determine ineffective software prefetches due to an L2 cache hit.

Suggested Workaround

None.

Fix Planned

No fix planned

538 Performance Counter Does Not Count for Some Retired Micro-Ops

Description

Some instructions with F0h in the opcode byte are incorrectly detected by the processor core as empty micro-ops, causing the processor core to not properly increment PMCx0C1. The following instructions may cause this performance monitor to undercount:

- FCOMI
- FCOMIP
- F2XM1

Potential Effect on System

Performance monitoring software will not have an accurate count of retired micro-ops. The performance counter may undercount and the error is directly proportional to the number of the instructions listed above.

Suggested Workaround

None.

Fix Planned

No fix planned

540 GART Table Walk Probes May Cause System Hang

Description

Probes that are generated for GART table walks may overflow internal queues and lead to a deadlock.

Potential Effect on System

System hang.

Suggested Workaround

System software that initializes the GART Table Base Address (D18F3x98) should set GART Aperture Control[DisGartTblWlkPrb] (D18F3x90[6]) = 1b. The GART tables should be in UC DRAM or be updated only using strongly-ordered uncacheable writes. System software should not set HWCR[ForceRdWrSzPrb] (MSRC001_0015[23]) if GART is enabled.

Fix Planned

No fix planned

550 Latency Performance Counters Are Not Accurate

Description

Latency performance counters NBPMCx1E2 through NBPMCx1E7 are not accurate when L3 speculative miss prefetching is enabled (D18F2x1B0[13] = 0b, Extended Memory Controller Configuration Low[SpecPrefDis]).

Potential Effect on System

Performance monitoring software cannot accurately measure latency events. The reported latency may greatly exceed the actual latency in some instances.

Suggested Workaround

No workaround is recommended. Performance monitoring software may set D18F2x1B0[13] = 1b to collect accurate latency values. This workaround has an impact to overall system performance.

Fix Planned

No fix planned

585 Incorrect Memory Controller Operation Due to a WrDatGrossDly Setting of 3.5 MEMCLKs

Description

The memory controller may incorrectly issue a ZQ command during a 64-byte write operation when WrDatGrossDly is set to a value of 3.5 MEMCLKs (111b).

Potential Effect on System

Undefined system behavior.

Suggested Workaround

If WrDatGrossDly (D18F2x9C_x0000_0[3:0]0[3:1]_dct[1:0]) for all byte lanes (including the ECC byte lane) and all populated DIMMs = 111b, BIOS should set DataTxFifoWrDly (D18F2x210_dct[1:0]_nbp[3:0] bits 18:16) as specified in the *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 00h-0Fh Processors*, order# 42301, but with a minimum value of 010b.

Fix Planned

No fix planned

586 A Far Control Transfer Changing Processor Operating Mode May Generate a False Machine Check

Description

A far control transfer that changes the processor operating mode may erroneously indicate a decoder instruction buffer parity error (DEIBP) machine check, leading to a system shutdown. The extended error code logged in the IF Machine Check Status register indicates a decode instruction buffer error (MSR0000_0405[20:16] = 10010b).

Potential Effect on System

Machine check exception due to a decoder instruction buffer parity error leading to system shutdown.

Suggested Workaround

BIOS should set MSRC001_0045[18] = 1b (MC1_CTL_MASK[DEIBP]).

Fix Planned

No fix planned

592 VPEXTRQ and VPINSRQ May Not Signal Invalid-Opcode Exception

Description

Advanced Vector Extensions (AVX) variants of legacy SSE instructions normally promote the size of a GPR operand using VEX.W. When running in 32-bit legacy or compatibility modes, setting VEX.W=1 is nonsensical and VEX.W is ignored. VPEXTRQ and VPINSRQ are an exception to that general rule and are specified to generate a Invalid-Opcode (#UD) exception. In violation of this, the processor does not signal #UD exception for AVX instructions VPEXTRQ and VPINSRQ when VEX.W=1 and the processor is running in 32-bit legacy or compatibility modes. Instead, the instruction is executed as if VEX.W=0.

Potential Effect on System

None expected. These opcode encodings are not expected to be generated by software.

Suggested Workaround

Software should only generate VPEXTRQ and VPINSRQ instructions with VEX.W=0 when operating in 32-bit modes and not depend on generating a #UD with VEX.W=1.

Fix Planned

No fix planned

593 Last-Branch Record Enabled May Cause Machine Check and Incorrect LastBranchToIp

Description

When LBR is enabled, a complex interaction between two threads of the same compute-unit may result in the processor core reporting an incorrect value in the LastBranchToIp register (MSR0000_01DC).

Potential Effect on System

In rare circumstances, the value reported in LastBranchToIP may present incorrect debug information. The processor may also report an uncorrectable machine check exception for a branch status register parity error, simultaneous to the above error. MC1_STATUS[ErrorCodeExt] (MSR0000_0405[20:16]) = 00110b identifies a branch status register parity error.

Suggested Workaround

BIOS should set MSRC001_0045[15] = 1b (MC1_CTL_MASK[BSRP]). This workaround does not resolve the potential for an incorrect address to be provided in LastBranchToIp. This latter effect has negligible impact on debugging due to the low probability of the error occurring when this data is being collected. No workaround is required for this aspect.

Fix Planned

No fix planned

600 HyperTransport™ Link Retry Due to Partial CRC Error May Cause System Hang

Description

The northbridge may stall when a probe hit returning data occurs simultaneously with a link retry due to a partial CRC error detected on an unrelated read packet. This error can only occur on a coherent HyperTransport™ link.

Potential Effect on System

System hang.

Suggested Workaround

BIOS should not alter D18F0x150[11:9] (Link Global Retry Control Register[HtRetryCrcDatIns]) from its reset value of 000b.

Fix Planned

No fix planned

602 HyperTransport™ Link Frequency Changes May Cause a System Hang

Description

A HyperTransport™ link operating at a Gen3 frequency (greater than 2.0 GT/s) may have excessive link retries or may fail to train after transitioning to a new Gen3 frequency. The failure is due to a misconfiguration of the HyperTransport Link Phy Receiver DLL Control and Test 5 Register [DllProcessFreqCtlOverride, DllProcessFreqCtlIndex2] settings (D18F4x1[9C,94,8C,84]_x[5:4][9:0][8,0]F_dm[1] bits 12 and 3:0). These settings are specific to each Gen3 frequency and may be programmed only while the link is operating at a Gen1 frequency (less than or equal to 2.0 GT/s).

Potential Effect on System

System hang.

Suggested Workaround

Software that transitions a HyperTransport link between two different Gen3 frequencies must first transition the link to a Gen1 frequency so that the DllProcessFreqCtlOverride and DllProcessFreqCtlIndex2 settings may be programmed according to the algorithm documented in the *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 00h-0Fh Processors*, order# 42301.

Fix Planned

No fix planned

608 P-state Limit Changes May Not Generate Interrupts

Description

P-state limit changes fail to generate interrupts when the target P-state limit is a higher or equal performance P-state (lower or equal numbered P-state) than the Application Power Management (APM) P-state limit.

Potential Effect on System

Operating systems monitoring processor P-state capabilities may not be notified of all P-state limit changes, resulting in either one of the following conditions:

- The processor runs continuously in a lower performance (higher numbered) P-state than is actually available.
- The operating system may request a higher performance (lower numbered) P-state than is actually available.

Suggested Workaround

BIOS should set MSRC001_1000[16] = 1b.

Fix Planned

No fix planned

619 Non-Posted Reads May Block Write Dependent on Probe Responses

Description

The northbridge may stall indefinitely on non-posted reads when a posted write becomes dependent on probe responses.

Potential Effect on System

System hang.

Suggested Workaround

BIOS should set D18F5x88[14] = 1b.

Fix Planned

No fix planned

623 Small Code Segment Limits May Cause Incorrect Limit Faults

Description

In cases where the code segment limit is less than 0_0020h and the Granularity (G) bit is zero , the processor reports an incorrect #GP exception even when no limit violation exists.

Potential Effect on System

None expected. In the unlikely case that a code segment is 32 bytes or smaller, an unexpected #GP exception may occur.

Suggested Workaround

None required. It is anticipated that code segment sizes are greater than 32 bytes.

Fix Planned

No fix planned

624 SB-RMI Processor State Accesses May Persistently Timeout if Interrupted by a Warm Reset

Description

The assertion of a warm reset during a small timing window of an APML SB-RMI processor state access may cause the internal processor state access interface to hang. A protocol status code of 11h or 12h (Command Timeout) is returned, however, the internal interface remains hung and all future SB-RMI processor state accesses receive command timeouts until a cold reset is performed. If SB-RMI timeouts are disabled (Control Register[TimeoutDis], SBRMI_x01[2]), the SB-RMI processor state accesses will not receive a successful completion, instead of a command timeout.

Potential Effect on System

Under rare circumstances, system management software will not be able to access processor state using SB-RMI processor state accesses. SB-TSI accesses and SB-RMI register accesses are not impacted. Refer to Advanced Platform Management Link (APML) Specification, order# 41918 for details on differentiating SB-RMI processor state accesses from SB-RMI register accesses.

Suggested Workaround

None required.

Fix Planned

No fix planned

625 SB-RMI Writes May Not Be Observed by Processor

Description

After a write using the APLM SB-RMI interface to either the Inbound Message Registers (SBRMI_x3[F:8]) or Software Interrupt Register (SBRMI_x40), the processor may observe the previous contents (as if the write did not occur) when reading these same registers using the SBI Address/Data registers (D18F3x1E8 and D18F3x1EC). The conditions under which this erratum may occur requires that message-triggered C1E is enabled (D18F3xD4[13] = 1b, Clock Power/Timing Control 0[MTC1eEn]) . The functionality of the SB-RMI interface is not otherwise affected .

Potential Effect on System

Software running on the processor is not able to properly receive messages from system management software using the SB-RMI interface.

Suggested Workaround

None. In the event that system management software needs to communicate with software running on the processor, an alternative mechanism should be used.

Fix Planned

No fix planned

636 Instruction Addresses Near Canonical Address Limit May Cause #GP Exception

Description

The processor may incorrectly generate a #GP exception when an instruction executes within a small window of the linear-memory address at the limit of canonical address space (0000_7FFF_FFFF_FFFFh: 0000_7FFF_FFFF_FFF2h) and multiple branch mis-predicts occur to a linear-memory address at the limit of canonical address space.

Potential Effect on System

In the unlikely event that the conditions for this erratum occur, an unexpected #GP exception may result in a program or system crash.

Suggested Workaround

None required.

Fix Planned

No fix planned

637 Processor Does Not Report the Correct DRAM Address for MCA Errors Within the CC6 Save Area

Description

While reporting an ECC machine check error in the core C6 (CC6) save area, the processor may store an internal address in MC4_ADDR (MSR0000_0412) instead of the physical DRAM address. The stored internal address can be uniquely identified, as it matches 000000FD_F7xxxxxh.

Potential Effect on System

Software may not be able to correctly interpret the machine check addresses for either corrected or uncorrected DRAM errors. As a result, it may fail to report the correct physical location of the error.

Suggested Workaround

When using the address in MC4_ADDR (MSR0000_0412) software should compare MC4_ADDR[47:24] with 00FDF7h. If it matches 00FDF7h, then the following algorithm can be used to correct the value from MC4_ADDR into the physical DRAM address that is in error.

1. Software first determines which node (identified by its node ID) reported the machine check. This is usually known to software that reads MC4_ADDR, (i.e. NodeReportingMca = CPUID Fn8000_001e_ECX[NodeId, bits 7:0]), but in some cases software may not know the node that reported the machine check. In this case, the node that reported the MCA can be determined as follows:
 - a. SourceNode = MC4_ADDR[22:20]. In this step, software determines the node that generated the CC6 save request. This is not necessarily the node that reported the machine check.
 - b. NodeReportingMca = D(18h+SourceNode)F4x128[14:12], where "18h+SourceNode" is the device number of the node that generated the CC6 save request. In this step, software accesses the C-state Policy Control 1 Register[CoreStateSaveDestNode] on the node that generated the CC6 save request (SourceNode from the previous step). This is the node that reported the machine check.
2. DramLimitSysAddrReg = D(18h+NodeReportingMca)F1x124, where "18h+NodeReportingMca" is the device number of the node that reported the machine check. In this step, software reads the register containing the DRAM Limit System Address from the node that reported the machine check. The register contents from this step are saved in a temporary variable for use in later steps.
3. Cc6BaseAddress[47:0] = {DramLimitSysAddrReg[20:0], ((DramLimitSysAddrReg[23:21] ^ 111b) << 24), 000000h}. In this step, software calculates the CC6 base address using the DramLimitAddr. DramLimitAddr[47:27] is bits 20:0 of the register read in step 2. Bits 26:24 of the CC6 base address is calculated from DramInlvEn.
4. NodeInterleavingEnabled = (DramLimitSysAddrReg[23:21] != 000b). In this step, software determines if node interleaving is enabled. Node interleaving is enabled if the register read in step 2 has a non-zero value in bits 23:21 (DramIntlvEn).
5. If node interleaving is not enabled (!NodeInterleavingEnabled from step 4):
 - a. The physical DRAM address of the machine check error is (Cc6BaseAddress + MC4_ADDR[23:0]) where Cc6BaseAddress is the result from step 3.
6. If node interleaving is enabled (NodeInterleavingEnabled from step 4):
 - a. TempMcaAddress = (MC4_ADDR[63:0] & 00000000_00FFF000h) << (log (base 2) of (DramLimitSysAddrReg[23:21] + 1)). In this step, a temporary variable is initialized using the value reported in MC4_ADDR, removing bits 47:24 and bits 11:0, and shifting the remaining bits by either 1 (two nodes), 2 (four nodes) or 3 (eight nodes). The node count is indirectly determined from DramIntlvEn (bits 23:21 of the register read in step 2).
 - b. DramBaseSysAddrReg = D(18h+NodeReportingMca)F1x120, where "18h+NodeReportingMca" is the device number of the node that reported the machine check. In this step, software reads the register

containing the DRAM Base System Address from the node that reported the machine check. The register contents from this step are saved in a temporary variable for use in the next step.

- c. $\text{TempMcaAddress} = \text{TempMcaAddress} | (\text{DramBaseSysAddrReg}[23:21] \ll 12)$. In this step, the `DramIntlvSel`, bits 23:21 of the register read in the previous step, is placed in bits 14:12 of the address.
- d. $\text{TempMcaAddress} = \text{TempMcaAddress} | \text{MC4_ADDR}[11:0]$. In this step, the low order bits of the machine check address are placed into the calculated address.
- e. The physical DRAM address of the machine check error is $(\text{Cc6BaseAddress} + \text{TempMcaAddress})$ where `Cc6BaseAddress` is the result from step 3.

The DRAM address that results from this workaround is only appropriate for determining the location of the failing memory. Software cannot access system DRAM using this address .

Fix Planned

No fix planned

657 MC1_STATUS Enable Bit Not Set When Logging Corrected Errors

Description

The processor does not set MC1_STATUS[En] = 1b (MSR0000_0405[60]) when logging an enabled and corrected error in the IF machine check register bank (bank 1). Software can identify the corrected errors that are affected by this erratum when it observes an MC1_STATUS register with all of the following:

- MC1_STATUS[Valid] (bit 63) = 1b
- MC1_STATUS[Uc] (bit 61) = 0b
- MC1_STATUS[En] (bit 60) = 0b
- MC1_STATUS[Pcc] (bit 57) = 0b
- The corresponding enable bit in MC1_CTL (MSR0000_0404) = 1b

Potential Effect on System

None expected.

Suggested Workaround

None required.

Fix Planned

No fix planned

658 CPUID Incorrectly Reports Large Page Support in L2 Instruction TLB

Description

The CPUID instruction incorrectly reports the number of entries and the associativity of 2 MB, 4 MB and 1 GB TLB entries in the L2 instruction TLB. The following CPUID fields are incorrectly zero:

- CPUID Fn8000_0006_EAX[L2ITlb2and4MSize]
- CPUID Fn8000_0006_EAX[L2ITlb2and4MAssoc]
- CPUID Fn8000_0019_EBX[L2ITlb1GSize]
- CPUID Fn8000_0019_EBX[L2ITlb1GAssoc]

Potential Effect on System

None expected.

Suggested Workaround

Software may substitute the following values to determine the number and associativity of large page L2 TLB entries:

- CPUID Fn8000_0006_EAX[L2ITlb2and4MSize] = 1024
- CPUID Fn8000_0006_EAX[L2ITlb2and4MAssoc] = 6
- CPUID Fn8000_0019_EBX[L2ITlb1GSize] = 1024
- CPUID Fn8000_0019_EBX[L2ITlb1GAssoc] = 6

This workaround should only be implemented after checking for affected silicon family and model (CPUID Fn0000_0001_EAX[Extended Family, Family, Extended Model and Model]), and that the processor reports zero for these fields.

Fix Planned

No fix planned

659 VMCB Interrupt Shadow Status May Be Incorrect

Description

The processor may fail to clear the VMCB INTERRUPT_SHADOW field (VMCB offset 068h bit 0) when intercepting or interrupting an SVM guest that is executing a Move String instruction with a REP prefix under interrupt shadow. This erratum does not occur on the last iteration of the Move String instruction.

Potential Effect on System

The SVM guest may continue to operate under interrupt shadow until the Move String instruction has completed. This may delay servicing of a pending interrupt.

Suggested Workaround

Contact your AMD representative for information on a BIOS update.

Fix Planned

No fix planned

660 APERF May Increase Unpredictably

Description

The value of MSR0000_00E8 (APERF) may increase unpredictably after any of the following events:

- A P-state transition, including those performed due to core performance boost (CPB).
- A C-state transition.
- A change in the P-state limit due to hardware thermal control (HTC), application power management (APM) or advanced platform management link (APML) TDP limiting.
- Execution of the MWAIT instruction.

Potential Effect on System

Software may calculate the effective frequency of a core incorrectly or observe that the APERF register value appears to increase unpredictably.

Suggested Workaround

Contact your AMD representative for information on a BIOS update.

Fix Planned

No fix planned

661 P-State Limit and Stop Clock Assertion May Cause System Hang

Description

A P-state limit change that occurs within a small timing window of a Stop Clock assertion may result in DRAM not entering self-refresh mode for an S3 sleep state transition, or a system hang if it occurs while another processor core is transitioning to the Core C6 (CC6) state.

Potential Effect on System

System hang.

Suggested Workaround

Contact your AMD representative for information on a BIOS update.

Fix Planned

No fix planned

663 Local Interrupts LINT0/LINT1 May Occur While APIC is Software Disabled

Description

The processor unmask local interrupts (LINT0 and LINT1) while the APIC is software disabled (Spurious-Interrupt Vector Register[APICSWEn], APICF0[8] = 0b). The LINT[1:0] LVT entry mask bits (APIC3[60:50] bit 16) are cleared and cannot be set. Broadcast ExtInt and NMI interrupt requests can be converted to LINT0 and LINT1 local interrupts respectively by setting Link Transaction Control Register[LintEn] (D18F0x68[16]) = 1b. If this bit is set while the APIC is software disabled, an ExtInt or NMI interrupt causes an unexpected local interrupt.

Potential Effect on System

Software may receive a local interrupt that was not expected, possibly leading to a system crash.

Suggested Workaround

BIOS should set MSRC001_001F[23] = 1b before enabling the APIC (APIC_BAR[ApicEn] (MSR0000_001B[11]) = 1b) or before setting Link Transaction Control[LintEn] (D18F0x68[16]) = 1b.

Fix Planned

No fix planned

667 Processor May Generate Incorrect P-state Limit Interrupts

Description

P-state limit changes due to SB-RMI (SBI P-state Limit[PstateLimit], MSRC001_0072[10:8]), software (Software P-state Limit Register[SwPstateLimit], D18F3x68[30:28]), or hardware thermal control (entering HTC-active state, i.e. PROCHOT# assertion) may generate duplicate interrupts when Hardware Thermal Control Register bits [PslApicLoEn, PslApicHiEn] are not both zero (D18F3x64[7:6] != 00b). The processor actually uses APM TDP Control[ApmTdpLimitIntEn] = 1b to enable the generation of interrupts for P-state limit changes due to SB-RMI, software, or HTC, as well as to generate interrupts for changes to TDP Limit 3 Register[ApmTdpLimit] (D18F5xE8[28:16]).

Potential Effect on System

Operating systems monitoring processor P-state capabilities may receive duplicate notification of P-state limit changes due to SB-RMI, software, or HTC.

Suggested Workaround

BIOS should leave Hardware Thermal Control[PslApicLoEn, PslApicHiEn] at their default reset value (D18F3x64[7:6] = 00b) and should set APM TDP Control[ApmTdpLimitIntEn] (D18F4x16C[4]) = 1b. This workaround requires software to receive both P-state limit change interrupts and ApmTdpLimit change interrupts .

Fix Planned

No fix planned

668 Load Operation May Receive Incorrect Data After Floating-point Exception

Description

The processor may incorrectly load data from a prior store-to-load forwarding operation after an unmasked x87 floating-point exception (#MF) if the exception occurs while CR0[Numeric Error] = 0b (CR0.NE, bit 5) and a prior exception is indicated (x87 Status Word Register[Exception Status], FSW.ES bit 7, is already 1b).

Potential Effect on System

None expected. Operating systems typically set CR0.NE = 1b and floating-point exception handlers normally clear the exception status (FSW.ES). If these conditions are not met, a load operation may receive data that was not updated by the most current write from a processor core.

Suggested Workaround

Contact your AMD representative for information on a BIOS update.

Fix Planned

No fix planned

671 Debug Breakpoint on Misaligned Store May Cause System Hang

Description

A misaligned store that crosses cache lines and requires an address translation due to a TLB miss may cause a system hang if the trailing cache line has an address breakpoint enabled through DR7.

Potential Effect on System

System hang.

Suggested Workaround

Contact your AMD representative for information on a BIOS update. This workaround has a performance impact when certain debug breakpoints are enabled. System developers that wish to enable debug breakpoints without this workaround may first set `MSRC001_1000[17] = 1b`. AMD recommends this workaround be enabled with AMD Opteron™ processors. For all other processors, BIOS should disable the workaround by setting `MSRC001_1000[17] = 1b`.

Fix Planned

No fix planned

672 SVM Guest Performance Counters May be Inaccurate Due to SMI

Description

Performance Event Counters[5:0] (MSRC001_020[B,9,7,5,3,1]) incorrectly count events in System-Management Mode (SMM) after a Secure Virtual Machine (SVM) guest receives a System-Management Interrupt (SMI) that is not intercepted by the host. This occurs when guest event counting is enabled by setting Performance Event Select[5:0][HostGuestOnly] = 01b (MSRC001_020[A,8,6,4,2,0][41:40]) and EFER[SVME] = 1b (MSRC000_0080[12]).

Potential Effect on System

Performance monitoring software overcounts events for an SVM guest when non-intercepted SMIs occur.

Suggested Workaround

Contact your AMD representative for information on a BIOS update. When the workaround is enabled, the processor swaps the HostGuestOnly bits (i.e. bits 41 and 40 of MSRC001_020[A,8,6,4,2,0] are exchanged) before entering SMM from SVM guest mode and again on the corresponding RSM.

Fix Planned

No fix planned

673 Misaligned Page Crossing String Operations May Cause System Hang

Description

A misaligned Move String or Store String instruction with a REP prefix that crosses a page boundary may cause a system hang. This may occur when the TLB entry for the leading page is evicted while the string operation is executing, or the trailing page of the string operation crosses into a large page (1 GB or 2 MB) which requires an address translation due to a TLB miss.

Potential Effect on System

Unpredictable system behavior, likely leading to a system hang.

Suggested Workaround

Contact your AMD representative for information on a BIOS update.

Fix Planned

No fix planned

674 Processor May Cache Prefetched Data from Remapped Memory Region

Description

Prefetches from a write back (WB) DRAM memory region may persist when that memory region is remapped to an uncacheable (UC) or write combining (WC) memory type.

Potential Effect on System

Data could be cached in a modified state from the remapped memory region, which will not be probed, however this can only occur if a prefetch operation persists through the invalidation or flushing of TLB entries and cache lines before the remapped memory region is accessible in a coherent manner. There have been no observations of this erratum on silicon.

Suggested Workaround

None recommended. Optionally, system software may set MSRC001_1022[13] = 1b (DC_CFG[DisHwPf]) during system boot if frequent run-time remapping of memory types as described is expected.

Fix Planned

No fix planned

675 Instructions Performing Read-Modify-Write May Alter Architectural State Before #PF

Description

An instruction performing a read-modify-write operation may be presented with a page fault (#PF) after modifying architectural state.

Potential Effect on System

The processor may present a #PF exception after some of the instruction effects have been applied to the processor state. No system effect is observed unless the operating system's page fault handler has some dependency on this interim processor state, which is not the case in any known operating system software. The interim state does not impact program behavior if the operating system resolves the #PF and resumes the instruction. However, this interim state may be observed by a debugger or if the operating system changes the #PF to a program error (for example, a segmentation fault).

Suggested Workaround

None recommended.

Fix Planned

No fix planned

689 AM3r2 Six Core Processor May Limit P-state When Core C6 State is Disabled

Description

If system software disables core C6 state (CC6) on an AMD FX™ 6100 Six-Core Processor, OPN FD6100WMW6KGU, the application power management (APM) incorrectly limits the processor P-state to a low performing P-state. Refer to the *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 00h-0Fh Processors*, order# 42301 section "Core C6 (CC6) State" for the conditions under which CC6 state is disabled.

Potential Effect on System

None expected under normal circumstances, as AMD recommends that CC6 is enabled in all systems. In the event that the system software implements an option to disable CC6, the system may experience low performance in this mode.

Suggested Workaround

AMD recommends that CC6 is enabled. This erratum does not apply in this case and no workaround is necessary. In the event that system software disables CC6, disable APM using Core Performance Boost Control Register[ApmMasterEn] (D18F4x15C[7]) = 0b. To avoid unnecessarily disabling APM on unaffected parts, the following conditions should be met before implementing this workaround:

- CC6 is disabled.
- CPUID Fn8000_0001_EBX[31:28] (BrandID Identifier[PkgType]) = 0001b (AM3r2 package type).
- CPUID Fn8000_0008_ECX[NC] (APIC ID Size and Core Count[Number of Physical Cores]) = 5.
- D18F4x150[7:0] = 6Bh.
- D18F4x10C[11:0] (TDP Limit 2 Register[NodeTdpLimit]) = 130h.

Fix Planned

No fix planned

690 Northbridge FIFO Read/Write Pointer Overlap May Cause Hang or Protocol Error Machine Check

Description

A command or data transfer may be lost when the write pointer overlaps the read pointer of a synchronization FIFO between the processor core and the northbridge.

Potential Effect on System

Unpredictable system behavior, likely leading to a hang of both cores of a compute-unit. The processor may also report a probe filter protocol machine check exception identified by the extended error code in the NB Machine Check Status Register (MC4_STATUS[ErrorCodeExt], MSR0000_0411[20:16] = 01011b).

Suggested Workaround

BIOS should set D18F3xDC[14:12] = 101b (Clock Power/Timing Control 2 [NbsynPtrAdj]). A warm reset is required before this setting takes effect.

Fix Planned

No fix planned

693 Performance Counter May Incorrectly Count MXCSR Loads

Description

The processor may incorrectly increment the following performance counter due to XRSTOR, FXRSTOR, LDMXCSR or VLDMXCSR instructions loading the MXCSR register:

- PMCx003 (Retired Floating Point Ops)

Potential Effect on System

Performance monitoring software will not have an accurate count of the number of retired floating point operations reported by the above performance counter.

Suggested Workaround

None.

Fix Planned

No fix planned

694 IBS Sampling of Instruction Fetches May be Uneven

Description

Instructions selected for instruction-based sampling (IBS) of fetch performance (Fetch Control[IbsFetchEn], MSRC001_1030[48] = 1b) may be sampled unevenly when the instruction fetch stream is redirected (e.g., due to a branch taken).

Potential Effect on System

Performance monitoring software may not receive even, unbiased IBS sampling of the instruction fetch stream. However, IBS can still be used effectively for identifying performance issues associated with specific instructions. The sampling bias makes IBS less effective for measuring the statistical distribution of operations and events.

Suggested Workaround

None.

Fix Planned

No fix planned

695 Processor May Interpret FCW Incorrectly After FNSAVE/ FSAVE Limit Fault

Description

The processor operates as if the floating-point control word (FCW) has been initialized after executing an FNSAVE or FSAVE instruction which generates a stack limit fault (SS). This occurs when the instruction attempts to store the state of the floating-point unit to a memory location that crosses a 16-bit (0xFFFF) or 32-bit (0xFFFF_FFFF) address boundary in real or protected mode respectively, and persists until software reinitializes the FCW. The FXSAVE instruction is not affected by this erratum.

Potential Effect on System

None expected during normal operation. A stack limit fault while executing an FNSAVE or FSAVE instruction is unusual and AMD has not observed the above conditions in any commercially available software. In the unlikely event that software creates the conditions described above one of the following may occur:

- The processor may write an indefinite value, as if masked, when signaling an invalid-operation exception (IE) after an FLD instruction executes with invalid operands while invalid operations are unmasked (FCW.IM, bit 0 = 0b).
- The processor may set the FERR signal incorrectly after an FLDCW instruction updates the floating-point control word mask bits (FCW[5:0]). A subsequent floating point operation may then result in an incorrect or missing x87 floating-point exception (#MF).

Suggested Workaround

None required.

Fix Planned

No fix planned

Documentation Support

The following documents provide additional information regarding the operation of the processor:

- *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 15h Models 00h-0Fh Processors*, order# 42301
- *AMD64 Architecture Programmer's Manual Volume 1: Application Programming*, order# 24592
- *AMD64 Architecture Programmer's Manual Volume 2: System Programming*, order# 24593
- *AMD64 Architecture Programmer's Manual Volume 3: General-Purpose and System Instructions*, order# 24594
- *AMD64 Architecture Programmer's Manual Volume 4: 128-Bit and 256-Bit Media Instructions*, order# 26568
- *AMD64 Architecture Programmer's Manual Volume 5: 64-Bit Media and x87 Floating-Point Instructions*, order# 26569
- *AMD CPUID Specification*, order# 25481
- *HyperTransport™ I/O Link Specification* (www.hypertransport.org)
- *Advanced Platform Management Link (APML) Specification*, order# 41918

See the AMD Web site at www.amd.com for the latest updates to documents.