



# Revision Guide for AMD Family 15h Models 10h-1Fh Processors

Publication # <b>48931</b>	Revision: <b>3.10</b>
Issue Date: <b>May 2013</b>	

*Advanced Micro Devices* 

© 2011-2013 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 discontinue or make changes to products, specifications, product descriptions or documentation at any time without notice. The information contained herein may be of a preliminary or advance nature. 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 Radeon, AMD Athlon, AMD Sempron, and combinations thereof, are trademarks of Advanced Micro Devices, Inc.

PCIe and PCI Express are registered trademarks of PCI-SIG.

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

---

### **Disclaimer**

While every precaution has been taken in the preparation of this document, Advanced Micro Devices, Inc. assumes no liability with respect to the operation or use of AMD hardware, software or other products and documentation described herein, for any act or omission of AMD concerning such products or this documentation, for any interruption of service, loss or interruption of business, loss of anticipatory profits, or for punitive, incidental or consequential damages in connection with the furnishing, performance, or use of the AMD hardware, software, or other products and documentation provided herein. Ensure that you have the latest documentation.

# List of Figures

Figure 1. Format of CPUID Fn0000\_0001\_EAX.....9

## List of Tables

Table 1. Arithmetic and Logic Operators.....	8
Table 2. CUID Values for AMD Family 15h Models 10h-1Fh FP2 Processor Revisions.....	9
Table 3. CUID Values for AMD Family 15h Models 10h-1Fh FS1r2 Processor Revisions.....	9
Table 4. CUID Values for AMD Family 15h Models 10h-1Fh FM2 Processor Revisions.....	10
Table 5. AMD Family 15h Graphic Device IDs.....	10
Table 6. Cross Reference of Product Revision to OSVW ID.....	13
Table 7. Cross-Reference of Processor Revision to Errata.....	14
Table 8. Cross-Reference of Errata to Package Type.....	17
Table 9. Cross-Reference of Errata to Processor Segments.....	19

## Revision History

---

Date	Revision	Description
May 2013	3.10	Updated erratum #746; Added errata #775, #778, #786-#788; Updated <a href="#">Graphic Device IDs</a> , Updated <a href="#">Documentation Support</a> section to remove deprecated CPUID specification (CPUID instruction information is now contained in <i>AMD64 Architecture Programmer's Manual Volume 3: General-Purpose and System Instructions</i> , order# 24594).
August 2012	3.08	Added AMD FirePro™ Accelerated Processing Unit with AMD Radeon HD Graphics to <a href="#">Overview</a> , <a href="#">Table 4</a> , and <a href="#">Table 9</a> ; Updated erratum #709; Added errata #740 and #746; Updated <a href="#">Documentation Support</a> section; Updated <a href="#">Table 8</a> .
July 2012	3.06	Added errata #734, #737 and #739; Updated <a href="#">Table 5</a> .
June 2012	3.02	Added AMD R-Series APU embedded brand to <a href="#">Overview</a> , <a href="#">Table 2</a> , <a href="#">Table 3</a> , and <a href="#">Table 9</a> ; Added AMD A-Series APU to <a href="#">Overview</a> and <a href="#">Table 4</a> ; Added FP2 and FM2 packages to <a href="#">Table 2</a> , <a href="#">Table 4</a> , <a href="#">Table 5</a> , and <a href="#">Table 8</a> .
May 2012	3.00	Initial public release.

## Overview

---

The purpose of the *Revision Guide for AMD Family 15h Models 10h-1Fh 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 A-Series Accelerated Processing Unit (APU) with AMD Radeon™ HD Graphics
- AMD A-Series Mobile Accelerated Processing Unit (APU) with AMD Radeon HD Graphics
- AMD R-Series Accelerated Processing Unit (APU) with AMD Radeon HD Graphics
- AMD R-Series Processor
- AMD FirePro™ Accelerated Processing Unit with AMD Radeon HD Graphics
- AMD Athlon™ Dual-Core Processor
- AMD Athlon Quad-Core Processor
- AMD Sempron™ Dual-Core 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 10h-1Fh Processors*, order# 42300. 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]). Some registers in D18F2xXXX have an `_mp[1:0]` mnemonic suffix, which indicates there is one instance per memory P-state. The memory P-state instance is selected by DCT Configuration Select[MemPsSel] (D18F1x10C[3]).
- **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 10h-1Fh Processors*, order# 42300 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]). Some registers in D18F2xXXX\_xZZZZZ have an `_mp[1:0]` mnemonic suffix, which indicates there is one instance per memory P-state. The memory P-state instance is selected by DCT Configuration Select[MemPsSel] (D18F1x10C[3]).
- **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 "raised 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).

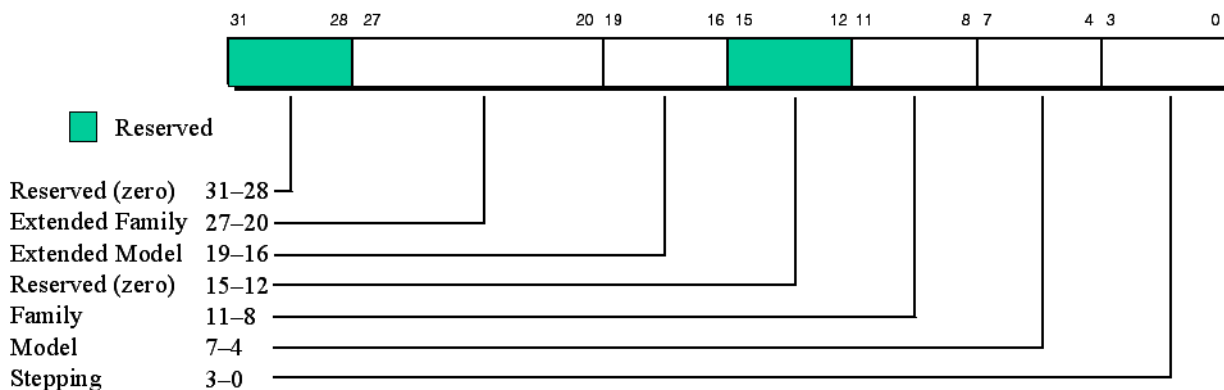


## Processor Identification

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 CUID instruction function 0000\_0001h (CUID Fn0000\_0001\_EAX). [Figure 1](#) shows the format of the value from CUID 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 D18F4x164 Fixed Errata Register (see [D18F4x164 Fixed Errata Register](#)).



**Figure 1. Format of CUID Fn0000\_0001\_EAX**

The following tables show the identification numbers from CUID 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. CUID Values for AMD Family 15h Models 10h-1Fh FP2 Processor Revisions**

CUID Fn0000_0001_EAX (Mnemonic)	AMD A-Series Mobile APU	AMD R-Series APU	AMD R-Series Processor
00610F01h (TN-A1)	X	X	X
00610F31h (RL-A1)	X	X	X

**Table 3. CPUID Values for AMD Family 15h Models 10h-1Fh FS1r2 Processor Revisions**

CPUID Fn0000_0001_EAX (Mnemonic)	AMD A-Series Mobile APU	AMD R-Series APU	AMD R-Series Processor
00610F01h (TN-A1)	X	X	X
00610F31h (RL-A1)	X	X	X

**Table 4. CPUID Values for AMD Family 15h Models 10h-1Fh FM2 Processor Revisions**

CPUID Fn0000_0001_EAX (Mnemonic)	AMD A-Series APU	AMD FirePro™ APU	AMD Athlon™ Dual-Core Processor	AMD Athlon™ Quad-Core Processor	AMD Sempron™ Dual-Core Processor
00610F01h (TN-A1)	X	X	X	X	X
00610F31h (RL-A1)	X	X	X	X	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 Fn0000\_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_0000h. Reserved.

## Graphic Device IDs

Processors with an integrated AMD Radeon HD Graphics Processing Engine use a graphics device ID at D1F0x00[31:16] to further identify the processor. [Table 5](#) lists the graphic device ID values in use for AMD Family 15h Models 10h-1Fh Accelerated Processing Units.

**Table 5. AMD Family 15h Graphic Device IDs**

D1F0x00[31:16]	FP2	FM2	FS1r2	Notes
9900h			X	Mobile (4 Quad Pipes, 6 SIMD& Texture Units, 2 Render Backends)
9901h		X		Desktop (4 Quad Pipes, 6 SIMD& Texture Units, 2 Render Backends)
9903h			X	Mobile (4 Quad Pipes, 4 SIMD&Texture Units, 2 Render Backends)
9904h		X		Desktop (4 Quad Pipes, 4 SIMD&Texture Units, 2 Render Backends)
9906h		X		Workstation (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
9907h	X			Mobile (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
9908h	X			Mobile (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
990Ah	X			Mobile (4 Quad Pipes, 4 SIMD&Texture Units, 2 Render Backends)
990Bh	X			Mobile (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
990Ch		X		Desktop (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
990Dh			X	Mobile (4 Quad Pipes, 4 SIMD&Texture Units, 2 Render Backends)
990Eh		X		Desktop (4 Quad Pipes, 4 SIMD&Texture Units, 2 Render Backends)
990Fh	X			Mobile (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
9910h			X	Embedded (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
9913h			X	Embedded (4 Quad Pipes, 4 SIMD&Texture Units, 2 Render Backends)
9917h	X			Embedded (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
9918h	X			Embedded (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
9919h	X			Embedded (4 Quad Pipes, 4 SIMD&Texture Units, 2 Render Backends)
9991h		X		Desktop (4 Quad Pipes, 3 SIMD&Texture Units, 1 Render Backend)
9993h		X		Desktop ( 4 Quad Pipes, 2 SIMD&Texture Units, 1 Render Backend)
9994h	X			Mobile (4 Quad Pipes, 3 SIMD&Texture Units, 1 Render Backend)
9995h			X	Mobile (4 Quad Pipes, 3 SIMD&Texture Units, 1 Render Backend)
9996h		X		Desktop (4 Quad Pipes, 3 SIMD&Texture Units, 1 Render Backend)
9997h			X	Mobile (4 Quad Pipes, 2 SIMD&Texture Units, 1 Render Backend)
9998h		X		Desktop (4 Quad Pipes, 2 SIMD&Texture Units, 1 Render Backend)
9999h	X			Mobile (4 Quad Pipes, 6 SIMD&Texture Units, 2 Render Backends)
999Ah	X			Mobile (4 Quad Pipes, 3 SIMD&Texture Units, 1 Render Backend)
999Bh	X			Mobile (4 Quad Pipes, 2 SIMD&Texture Units, 1 Render Backend)
99A4h	X			Embedded (4 Quad Pipes, 3 SIMD&Texture Units, 1 Render Backend)

## 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 10h-1Fh Processors*, order# 42300 , 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 10h-1Fh Processors*, order# 42300 , 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 forms the name string 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 0005h prior to hand-off to the OS.

Bits	Description
63:16	Reserved.
15:0	<b>OSVW_ID_Length</b> : 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
63:5	<b>OsvwStatusBits</b> : Reserved. OS visible work-around status bits. Read-write.
4	<b>OsvwId4</b> : 1 = Hardware contains erratum #724, an OS workaround may be applied if available; 0 = Hardware has corrected erratum #724.
3	<b>OsvwId3</b> : Reserved, must be zero..
2	<b>OsvwId2</b> : Reserved, must be zero..
1	<b>OsvwId1</b> : Reserved, must be zero..
0	<b>OsvwId0</b> : Reserved, must be zero..

BIOS shall program the state of the valid status bits as shown in [Table 6](#) prior to hand-off to the OS.

**Table 6. Cross Reference of Product Revision to OSVW ID**

CPUID Fn0000_0001_EAX (Mnemonic)	MSRC001_0141 Bits
00610F01h (TN-A1)	0000_0000_0000_0010h

## 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. "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 7. Cross-Reference of Processor Revision to Errata**

No.	Errata Description	CPUID Fn0000_0001_EAX
		00610F01h (TN-A1)
361	Breakpoint Due to an Instruction That Has an Interrupt Shadow May Be Lost	No fix planned
470	Warm Reset May Cause System Hang	No fix planned
608	P-state Limit Changes May Not Generate Interrupts	No fix planned
625	SB-RMI Writes May Not Be Observed by Processor	No fix planned
638	Processor May Violate Trp During Dynamic Mode Switch	No fix planned
657	MC1_STATUS Enable Bit Not Set When Logging Corrected Errors	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
671	Debug Breakpoint on Misaligned Store May Cause System Hang	No fix planned
685	Some Processor Cores May Have Inaccurate Instruction Cache Fetch Performance Counter	No fix planned
686	Processor Does Not Implement MSRC001_0055	No fix planned
687	Memory Arbitration May Stall	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
697	P-State Limit Removal During PC6 Entry May Cause System Hang	No fix planned
698	Processor May Not Recognize P-state Limit Changes During Northbridge P-state Transition	No fix planned
699	Processor May Generate Illegal Access in VMLOAD or VMSAVE Instruction	No fix planned
703	SB-RMI Access While Northbridge is Power Gated May Cause Interface Hang	No fix planned
704	Processor May Report Incorrect Instruction Pointer	No fix planned

**Table 7. Cross-Reference of Processor Revision to Errata (continued)**

No.	Errata Description	CPUID Fn0000_0001_EAX
		00610F01h (TN-A1)
707	Performance Counter for Locked Operations May Count Cycles from Non-Locked Operations	No fix planned
709	Processor May Be Limited to Minimum P-state After a P-state Limit Change	No fix planned
712	Processor May Hang During Graphics Memory Controller Sequencing	No fix planned
713	Processor May Hang Due to Short Graphics Memory Controller Activity	No fix planned
715	Processor May Alter Machine Check Registers on a Warm Reset	No fix planned
716	Warm Reset After BIOS Changes P-state During Early Boot May Cause System Hang	No fix planned
718	Instruction-Based Sampling May Be Inaccurate	No fix planned
719	Instruction-Based Sampling Fetch Counter Always Starts at Maximum Value	No fix planned
720	Processor May Not Respect Interrupt Shadow	No fix planned
724	Unintercepted Halt Instruction May Cause Protocol Machine Check or Unpredictable System Behavior	No fix planned
725	Incorrect APIC Remote Read Behavior	No fix planned
726	Processor May Report Incorrect MCA Address for Loads that Cross Address Boundaries	No fix planned
728	IOMMU Event Not Flagged when DTE Reserved Bits Are Not Zero	No fix planned
729	IOMMU Event Log Not Generated for Invalid DTE GCR3 Table Root Pointer	No fix planned
730	IOMMU Event Log Not Generated for Not Present Host Intermediate Page Tables	No fix planned
731	Incorrect Translation with IOMMU v1 512 GB Page Table	No fix planned
732	IOMMU Event Log Ordering Violation	No fix planned
733	IOMMU PPR Log Ordering Violation	No fix planned
734	Processor May Incorrectly Store VMCB Data	No fix planned
737	Processor Does Not Check 128-bit Canonical Address Boundary Case on Logical Address	No fix planned
739	Processor May Read Branch Status Register With Inconsistent Parity Bit	No fix planned
740	Lightweight Profiling May Cause System Hang with Concurrent Stop Clock	No fix planned
746	IOMMU Logging May Stall Translations	No fix planned
775	Processor May Present More Than One #DB Exception on REP-INS or REP-OUTS Instructions	No fix planned
778	Processor Core Time Stamp Counters May Experience Drift	No fix planned
786	APIC Timer Periodic Mode is Imprecise	No fix planned
787	IOMMU Interrupt May Be Lost	No fix planned

**Table 7. Cross-Reference of Processor Revision to Errata (continued)**

No.	Errata Description	CPUID Fn0000_0001_EAX
		00610F01h (TN-A1)
788	Processor Incorrectly States Support for IOMMU Prefetch Command	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 8. Cross-Reference of Errata to Package Type**

Errata	Package		
	FM2	FP2	FS1r2
361	X	X	X
470	X	X	X
608	X	X	X
625	X	X	X
638	X	X	X
657	X	X	X
661	X	X	X
663	X	X	X
667	X	X	X
671	X	X	X
685	X	X	X
686	X	X	X
687	X	X	X
693	X	X	X
694	X	X	X
695	X	X	X
697	X	X	X
698	X	X	X
699	X	X	X
703	X	X	X
704	X	X	X
707	X	X	X
709	X	X	X
712	X	X	X
713	X	X	X
715	X	X	X
716	X	X	X
718	X	X	X
719	X	X	X
720	X	X	X

**Table 8. Cross-Reference  
of Errata to Package Type  
(continued)**

Errata	Package		
	FM2	FP2	FS1r2
724	X	X	X
725	X	X	X
726	X	X	X
728	X		
729	X		
730	X		
731	X		
732	X		
733	X		
734	X	X	X
737	X	X	X
739	X	X	X
740	X	X	X
746	X		
775	X	X	X
778	X	X	X
786	X	X	X
787	X		
788	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 9. Cross-Reference of Errata to Processor Segments**

Errata	Processor Segment							
	AMD A-Series APU	AMD A-Series Mobile APU	AMD Athlon Dual-Core Processor	AMD Athlon Quad-Core Processor	AMD FirePro™ APU	AMD R-Series APU	AMD R-Series Processor	AMD Sempron Dual-Core Processor
361	X	X	X	X	X	X	X	X
470	X	X	X	X	X	X	X	X
608	X	X	X	X	X	X	X	X
625	X	X	X	X	X	X	X	X
638	X	X	X	X	X	X	X	X
657	X	X	X	X	X	X	X	X
661	X	X	X	X	X	X	X	X
663	X	X	X	X	X	X	X	X
667	X	X	X	X	X	X	X	X
671	X	X	X	X	X	X	X	X
685	X	X	X	X	X	X	X	X
686	X	X	X	X	X	X	X	X
687	X	X	X	X	X	X	X	X
693	X	X	X	X	X	X	X	X
694	X	X	X	X	X	X	X	X
695	X	X	X	X	X	X	X	X
697	X	X	X	X	X	X	X	X
698	X	X	X	X	X	X	X	X
699	X	X	X	X	X	X	X	X
703	X	X	X	X	X	X	X	X
704	X	X	X	X	X	X	X	X
707	X	X	X	X	X	X	X	X
709	X	X	X	X	X	X	X	X
712	X	X	X	X	X	X	X	X

**Table 9. Cross-Reference of Errata to Processor Segments  
(continued)**

Errata	Processor Segment							
	AMD A-Series APU	AMD A-Series Mobile APU	AMD Athlon Dual-Core Processor	AMD Athlon Quad-Core Processor	AMD FirePro™ APU	AMD R-Series APU	AMD R-Series Processor	AMD Sempron Dual-Core Processor
713	X	X	X	X	X	X	X	X
715	X	X	X	X	X	X	X	X
716	X	X	X	X	X	X	X	X
718	X	X	X	X	X	X	X	X
719	X	X	X	X	X	X	X	X
720	X	X	X	X	X	X	X	X
724	X	X	X	X	X	X	X	X
725	X	X	X	X	X	X	X	X
726	X	X	X	X	X	X	X	X
728	X		X	X	X			X
729	X		X	X	X			X
730	X		X	X	X			X
731	X		X	X	X			X
732	X		X	X	X			X
733	X		X	X	X			X
734	X	X	X	X	X	X	X	X
737	X	X	X	X	X	X	X	X
739	X	X	X	X	X	X	X	X
740	X	X	X	X	X	X	X	X
746	X		X	X	X			X
775	X	X	X	X	X	X	X	X
778	X	X	X	X	X	X	X	X
786	X	X	X	X	X	X	X	X
787	X		X	X	X			X
788	X		X	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

## 470 Warm Reset May Cause System Hang

---

### Description

The processor may hang if a warm reset occurs while a register access to any of the PCI Express® controllers' internal registers is in progress. The processor may perform register accesses for internal management purposes that are transparent to software.

### Potential Effect on System

System hang.

### Suggested Workaround

System BIOS should set D0F0xE4\_x013[3:0]\_8063 bits 4, 5, 12, 13 and 14 to 1b. This must be done early in the BIOS boot sequence to minimize the possibility of a hang due to a warm reset during the boot sequence.

During a link reconfigure operation, system BIOS must perform the following steps:

1. Clear D0F0xE4\_x013[3:0]\_8063 bits 4, 5, 12, 13 and 14 to 0b.
2. Perform the link reconfigure operation.
3. Set D0F0xE4\_x013[3:0]\_8063 bits 4, 5, 12, 13 and 14 to 1b.

### 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

## 625 SB-RMI Writes May Not Be Observed by Processor

---

### Description

After a write using the APML 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 northbridge clock gating is enabled for any C-state action (C-state Control[NbClkGate2, NbClkGate1, NbClkGate0], D18F4x118[27, 11], D18F4x11C[11]). 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



## 638 Processor May Violate Trp During Dynamic Mode Switch

---

### Description

The processor may violate the precharge time (Trp) for a DIMM when sending a mode register set (MRS) command to dynamically adjust MR0[PPD] during a precharge power down.

This erratum may occur only when fast exit/slow exit (dynamic) mode is selected as follows:

- D18F2x94\_dct[1:0][15] = 1b (DRAM Configuration High[PowerDownEn])
- D18F2x84\_dct[1:0][23] = 1b (DRAM MRS[PchgPDModeSel])

### Potential Effect on System

Unpredictable system operation.

### Suggested Workaround

If D18F2x84\_dct[1:0] bit 23 (PchgPDModeSel) = 1b and D18F2x94\_dct[1:0] bit 15 (PowerDownEn) = 1b, then precharge time (D18F2x200\_dct[1:0]\_mp[1:0] bits 20:16, Trp) should be set one higher than the DIMM specified value.

### 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

## 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.

### 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.

### **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 instead set APM TDP Control[ApmTdpLimitIntEn] (D18F4x16C[4]) = 1b. In addition, BIOS should set D18F4x16C[1] = 1b and D18F5xAC[3] = 1b.

### **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 using 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. AMD recommends that BIOS disables this workaround by setting MSRC001\_1000[17] = 1b.

### Fix Planned

No fix planned

## 685 Some Processor Cores May Have Inaccurate Instruction Cache Fetch Performance Counter

---

### Description

The processor may over-report PMCx080 (instruction cache fetches) when the performance monitor is enabled on an odd processor core number (APIC20[ApicId] is odd - i.e. bit 24 is 1b), as compared to when the performance counter is used on an even processor core number.

### Potential Effect on System

Performance monitoring software may not have an accurate count of instruction cache fetch operations. The performance counter may overcount.

### Suggested Workaround

None.

### Fix Planned

No fix planned

---

## **686 Processor Does Not Implement MSRC001\_0055**

---

**Description**

The processor does not properly allow writes to MSRC001\_0055 (Interrupt Pending Register). A write to MSRC001\_0055 is ignored and a read to the register returns zero.

**Potential Effect on System**

BIOS is unable to program this register.

**Suggested Workaround**

Contact your AMD representative for information on a BIOS update.

**Fix Planned**

No fix planned



## 687 Memory Arbitration May Stall

---

### Description

The processor arbitration between CPU, I/O transactions and graphics transactions to the DRAM controller may not recognize and properly prioritize a CPU or I/O transaction that has been marked high priority. This causes the internal transaction arbiter to stall. This stall may occur only if the processor reports D18F3x1FC[29] = 0b (Product Information Register 1[EnDcqChgPriToHigh]).

### Potential Effect on System

The memory bus may stall for excessive periods, causing display buffer under-run (screen corruption).

### Suggested Workaround

BIOS should set D18F2x408\_dct[1:0] bit 0 (GMC to DCT Control 2[CpuElevPrioDis]) = !D18F3x1FC[29] (the inverse of D18F3x1FC bit 29) and should set D18F5x88[14] = D18F3x1FC[29] (without inverting).

### 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

## 697 P-State Limit Removal During PC6 Entry May Cause System Hang

---

### Description

The processor may hang if the following conditions occur:

- A P-state limit has been applied that is actively restricting the current P-state to a lower-performing P-state than is requested by software. This limit may be due to application power management (APM), hardware thermal control (HTC), or Software P-state Limit Register[SwPstateLimit] (D18F3x68[30:28]).
- The P-state limit is an equal or lower performance (higher numbered) P-state than the pop-down P-state (Pop Up and Down P-states Register[PopDownPstate], D18F3xA8[31:29]).
- The P-state limit is removed during the period of time that the processor enters package C6 (PC6) state, i.e., the processor has a new P-state limit that is a lower P-state number (higher performing) than the previously applied P-state limit.

### Potential Effect on System

System hang. While the conditions for the erratum occur during the entry to PC6, the hang is observed during the exit from PC6.

### Suggested Workaround

Contact your AMD representative for information on a BIOS update.

### Fix Planned

No fix planned

## 698 Processor May Not Recognize P-state Limit Changes During Northbridge P-state Transition

---

### Description

During the period of time that a northbridge P-state transition is in process, a concurrent core P-state limit change may not transition the core P-state to reflect this new limit. The core P-state limit change may be due to application power management (APM), hardware thermal control (HTC), or Software P-state Limit Register[SwPstateLimit] (D18F3x68[30:28]).

### Potential Effect on System

If the core P-state limit changed to a higher P-state limit (i.e., the core P-state is now limited to lower performing P-states), then the processor cores may continue to operate at a P-state that is higher performing than this limit. This may result in a violation of the appropriate thermal design power (TDP).

If the core P-state limit changed to a lower P-state limit (i.e., the core P-state is no longer limited to lower performing P-states), then the processor cores may continue to operate at the lower performing P-states.

In addition, the processor does not generate an APIC330 interrupt for this P-state limit change.

### Suggested Workaround

Contact your AMD representative for information on a BIOS update.

### Fix Planned

No fix planned

## 699 Processor May Generate Illegal Access in VMLOAD or VMSAVE Instruction

---

### Description

The processor may generate a speculative access during execution of a VMLOAD or VMSAVE instruction. The memory type used for this access is defaulted to WB DRAM memory type, however the address used may not be a valid DRAM address or it may be an address that is not specified as cacheable in the memory type (i.e., the actual memory type is UC or WC).

### Potential Effect on System

When the address is not a valid DRAM address, the processor may recognize a northbridge machine check exception for a link protocol error. This machine check exception causes a sync flood and system reset under AMD recommended BIOS settings. The machine check has the following signature:

- The MC4\_STAT register (MSR0000\_0411) is equal to BA000020\_000B0C0F. Bit 62 (error overflow) or bit 59 (miscellaneous valid) of MC4\_STAT may or may not be set.
- Bits 5:1 of the MC4\_ADDR register (MSR0000\_0412) is equal to 01001b, indicating that a coherent-only packet was issued to a non-coherent link.

When the address is actually a non-cacheable memory type, the processor may incorrectly cache the data, resulting in unpredictable system behavior.

AMD has only observed a northbridge link protocol error machine check. The incorrect caching of an uncacheable memory region has not been observed by AMD.

### Suggested Workaround

Contact your AMD representative for information on a BIOS update.

### Fix Planned

No fix planned

## 703 SB-RMI Access While Northbridge is Power Gated May Cause Interface Hang

---

### Description

The processor SB-RMI state machine may hang if a SB-RMI access is performed while the northbridge is power gated. The northbridge is power gated when all cores are in the core C6 (CC6) state and the applied C-state Control Register northbridge power gating bit (NbPwrGate[2:0]) is set to 1b (i.e., D18F4x11C[11], D18F4x118[26], or D18F4x118[11] is set to 1b).

### Potential Effect on System

SB-RMI accesses may hang, resulting in failures or timeouts being returned to the platform management software. A warm reset may be necessary for SB-RMI accesses to be restored.

### Suggested Workaround

None.

### Fix Planned

No fix planned



---

## 704 Processor May Report Incorrect Instruction Pointer

---

**Description**

Under a highly specific and detailed set of internal timing conditions, the processor may store an incorrect instruction pointer (rIP) while processing an interrupt or a debug trap exception (#DB).

**Potential Effect on System**

Unpredictable system behavior.

**Suggested Workaround**

Contact your AMD representative for information on a BIOS update.

**Fix Planned**

No fix planned

## 707 Performance Counter for Locked Operations May Count Cycles from Non-Locked Operations

---

### Description

PMCx024[2] may include cycles spent performing non-locked operations.

### Potential Effect on System

Performance monitoring software may receive an incorrect (larger) count of the number of cycles spent in the non-speculative phase of locked operations.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## 709 Processor May Be Limited to Minimum P-state After a P-state Limit Change

---

### Description

Following a change to the P-state limit or a core C6 (CC6) exit, the processor may incorrectly restrict the processor to the lowest-performing P-state (Clock Power/Timing Control 2 Register[HwPstateMaxVal], D18F3xDC[10:8]). This restriction may not match any of the actual P-state limits and does not get removed until a processor reset occurs.

P-state limit changes that may cause this erratum may be 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).

### Potential Effect on System

Processor performance is limited to the lowest-performing P-state.

### Suggested Workaround

Contact your AMD representative for information on a BIOS update.

### Fix Planned

No fix planned

## 712 Processor May Hang During Graphics Memory Controller Sequencing

---

### Description

The processor may hang during a graphics memory controller (GMC) sleep state transitioning. The failure may be processor specific and may be sensitive to temperature.

### Potential Effect on System

System hang.

### Suggested Workaround

BIOS should set D18F2x408\_dct[1:0] bit 31 = 1b.

### Fix Planned

No fix planned

## 713 Processor May Hang Due to Short Graphics Memory Controller Activity

---

### Description

The processor may hang if LCLK deep sleep is enabled (LCLK\_DEEP\_SLEEP\_CNTL[EnableDs], GMMx7A0[31] is 1b) and a short burst of activity occurs from the graphics memory controller.

### Potential Effect on System

System hang.

### Suggested Workaround

BIOS should set GMMx7A0[31] = 0b.

### Fix Planned

No fix planned

# 715 Processor May Alter Machine Check Registers on a Warm Reset

---

## Description

The processor may alter warm-reset persistent bits in MC0\_STATUS (MSR0000\_0401), MC0\_ADDR (MSR0000\_0402), MC0\_MISC (MSR0000\_0403), MC2\_STATUS (MSR0000\_0409), MC2\_ADDR (MSR0000\_040A), MC2\_MISC (MSR0000\_040B), MC5\_STATUS (MSR0000\_0415), MC5\_ADDR (MSR0000\_0416), MC5\_MISC (MSR0000\_0417), MC6\_STATUS (MSR0000\_0419) and MC6\_ADDR (MSR0000\_040A) during a warm reset. The status bits may be cleared of an actual error indication, or may be written to a non-zero value that does not correlate to an actual machine check error.

A machine check exception (#MC) is not created when these registers are incorrectly altered.

This erratum is only observed when the processor core operating frequency is less than 900 MHz at the time of the warm reset.

## Potential Effect on System

The operating system software may place the incorrect machine check status information in an error log.

In the event that the warm-reset was the result of an actual machine check, the processor may also over-write the machine check information and the reason for the warm reset is not available for diagnostic purposes.

## Suggested Workaround

Do not operate the processor core frequency less than 900 MHz. Software should not modify any of the P-state[7:0] Registers (MSRC001\_00[6B:64]) such that the CpuFid (bits 5:0) and the CpuDid (bits 8:6) specify a core operating frequency that is less than 900 MHz.

Some processors have a default configuration where the hardware thermal control (HTC) P-state is configured for a core frequency of 800 MHz. BIOS should perform the following algorithm to adjust this configuration:

1. Read Hardware Thermal Control Register[HtcPstateLimit] (D18F3x64[30:28])
2. Use this value to index into the P-state Registers to the P-state register used in HTC mode (MSR(C001\_0064 + HtcPstateLimit)).
3. Read this MSR and check if bits 8:6 (CpuDid) is 1h and bits 5:0 (CpuFid) is 0h. If these values are not observed, the algorithm ends at this step.
4. Read the MSR of the next higher-performing P-state register (MSR(C001\_0064 + HtcPstateLimit - 1)), and save the value for CpuVid (bits 16:9).
5. Write the HTC P-state MSR (MSR(C001\_0064 + HtcPstateLimit)) with CpuFid = 2h and the saved CpuVid from the previous step. The other fields in this MSR are unchanged.

The above algorithm should be run by BIOS before entering CC6 mode and before enabling hardware thermal control (before writing Hardware Thermal Control Register[HtcEn] (D18F3x64[0]) = 1b).

## Fix Planned

No fix planned

## 716 Warm Reset After BIOS Changes P-state During Early Boot May Cause System Hang

---

### Description

A warm reset that occurs following a BIOS change to the P-state may cause a system hang. In order for this erratum to occur, the BIOS change to the P-state must occur while at least one processor core has the corresponding core enable (CpuEn, D18F0x1DC[7:1]) bit clear after a reset.

### Potential Effect on System

System hang.

### Suggested Workaround

The system BIOS should not perform a P-state change before the core enable bits are set for all cores.

### Fix Planned

No fix planned

## 718 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 was a recently executed store instruction whose store address matches the load address in bits 11:0.
- 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.
- The processor may infrequently report an incorrect instruction pointer in the IBS Fetch Linear Address (MSRC001\_1031).

### 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 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.
- Performance monitoring software should not rely on MSRC001\_1031.

### Fix Planned

No fix planned



## 719 Instruction-Based Sampling Fetch Counter Always Starts at Maximum Value

---

### Description

When setting IBS Fetch Control Register[IbsFetchEn] = 1b to enable IBS fetch sampling, the periodic fetch counter always starts at the maximum value programmed into IBS Fetch Control Register[IbsFetchMaxCnt] (MSRC001\_1030[15:0]) instead of starting at the value written into IBS Fetch Control Register[IbsFetchCnt] (MSRC001\_1030[19:4]).

### Potential Effect on System

System software that is managing multiple processes or virtual machines with different IBS configurations may create unintended delays before the next IBS sample by writing to MSRC001\_1030.

In the event that system software consistently writes to MSRC001\_1030, it is possible that the IBS fetch counter never expires and no instruction fetches are tagged. AMD has not observed this effect with production software.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## 720 Processor May Not Respect Interrupt Shadow

---

### Description

Under a highly specific and detailed set of internal timing conditions, a #DB exception may be presented during execution of an instruction that is in an interrupt shadow. In order for this erratum to occur, the other processor core in the compute-unit must be performing microcoded functions that are uncommon in usage.

### Potential Effect on System

Under rare circumstances, a debug exception may occur in an interrupt shadow. Under common software use, this exception does not have a system effect. In the event that system software uses "STI, RET" instead of a single IRET instruction, or changes the stack segment simultaneously with the stack pointer (i.e. not using a flat segment for the stack), unpredictable system failure may result. AMD has not observed this erratum with any commercially available software.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## 724 Unintercepted Halt Instruction May Cause Protocol Machine Check or Unpredictable System Behavior

---

### Description

An unintercepted halt instruction executed in guest mode may result in a processor core being in a cache-flush-on-halt state while having VMCB data in the cache.

### Potential Effect on System

Northbridge machine check exception (#MC) for a link protocol error. This machine check exception causes a sync flood and system reset under AMD recommended BIOS settings. The machine check has the following signature:

- The MC4\_STAT register (MSR0000\_0411) is equal to BA000020\_000B0C0F. Bit 62 (error overflow) or bit 59 (miscellaneous valid) of MC4\_STAT may or may not be set.
- Bits 5:1 of the MC4\_ADDR register (MSR0000\_0412) is equal to one of 10011b, 10100b, 11000b or 11001.

In addition, it is possible for unpredictable system operation to occur without a machine check exception. For example, a processor core may not observe a write that is performed by another processor core. AMD has not observed this effect in any commercially available software.

### Suggested Workaround

Hypervisors should intercept HLT instructions by setting VMCB.Intercept\_HLT (offset 00Ch bit 24) to 1b.

### Fix Planned

No fix planned

## 725 Incorrect APIC Remote Read Behavior

---

### Description

The processor may provide incorrect APIC register data on an APIC remote register read. A remote read is performed using Interrupt Command Register Low[MsgType] of 011b (APIC300[10:8]). The processor may, but does not always, provide an error indication in the remote read status field (APIC300[17:16]).

This erratum does not impact the use of remote APIC reads by BIOS during early power-on-self-test (POST) when the remote read is performed for addresses APIC300-APIC3F0.

### Potential Effect on System

None expected, as it is anticipated that no software other than BIOS uses remote APIC reads.

### Suggested Workaround

Software should not use remote APIC reads.

### Fix Planned

No fix planned

## 726 Processor May Report Incorrect MCA Address for Loads that Cross Address Boundaries

---

### Description

In the event that a line fill error or system read data error is reported for some, but not all, bytes of an unaligned load instruction that crosses a cache line boundary (64 bytes), the processor may intermittently report the address of the unaffected cache line in MC0\_ADDR (MSR0000\_0402).

### Potential Effect on System

None expected.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## 728 IOMMU Event Not Flagged when DTE Reserved Bits Are Not Zero

---

### Description

The processor IOMMU may not generate an event log entry when a translation table walk encounters a valid device table entry (DTE) with reserved bits 8:2 not equal to 000\_0000b. The reserved bits are ignored by the processor.

### Potential Effect on System

None expected in the absence of IOMMU host software programming errors. In the presence of IOMMU host software programming errors, the device may receive and report a completion-abort response without an IOMMU event log to diagnose the reason for the completion-abort response. Debug information presented by the processor in the event logs may be missing.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## 729 IOMMU Event Log Not Generated for Invalid DTE GCR3 Table Root Pointer

---

### Description

The processor IOMMU does not generate an event log entry when a translation table walk encounters a valid device table entry (DTE) whose GCR3 table root pointer is not a valid system address. When the IOMMU is operating in nested paging mode, the erratum occurs only if the host page table translations of the GCR3 root pointer translates this address to an invalid system address.

Instead, the processor treats this as an IOMMU page fault.

### Potential Effect on System

None expected in the absence of IOMMU host software programming errors. In the presence of IOMMU host software programming errors, debug information presented by the processor in the event logs may be missing or incorrect.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## 730 IOMMU Event Log Not Generated for Not Present Host Intermediate Page Tables

---

### Description

The processor IOMMU, when performing a nested address translation of a guest virtual address to a host physical address, does not generate event logs in the case that the table walk encounters an intermediate (non-leaf) host page table that is not present (valid bit is 0b). The IOMMU does abort this translation request and properly signals an I/O page fault.

This erratum does not impact IOMMU operation with peripherals that use address translation service (ATS) in conjunction with peripheral page service requests (PPR). When PPR is not used, the page tables used by the IOMMU translation typically requires "pinned" pages where the present or valid bit is always 1b.

### Potential Effect on System

None expected in the absence of IOMMU or memory management software programming errors in the hypervisor or driver software errors in the guest. In the presence of programming errors, the device may receive and report a completion-abort response without an IOMMU event log to diagnose the reason for the completion-abort response. Debug information presented by the processor in the event logs may be missing.

### Suggested Workaround

None.

### Fix Planned

No fix planned



## **731 Incorrect Translation with IOMMU v1 512 GB Page Table**

### **Description**

The processor may perform incorrect IOMMU translations using an IOMMU v1 page table that has a page size of 512 GB.

### **Potential Effect on System**

None expected. In the unlikely event that IOMMU software uses a 512 GB sized page table, unpredictable system behavior may result. AMD has not observed this erratum with any commercially available software.

### **Suggested Workaround**

IOMMU software should limit the IOMMU v1 page table such that the page size is never greater than the system memory.

### **Fix Planned**

No fix planned

## 732 IOMMU Event Log Ordering Violation

---

### Description

The processor IOMMU does not maintain producer-consumer ordering between the IOMMU event log DMA writes and IOMMU MMIO register read completions. The processor core may read stale or uninitialized event logs from memory when a read response from the event log tail pointer register passes the corresponding event log DMA write. A series or burst of event log DMA writes would normally be necessary for this ordering violation to be observed.

### Potential Effect on System

Software may process an event log before it has been completely written, possibly resulting in the operating system or hypervisor taking improper corrective actions.

### Suggested Workaround

The IOMMU driver of the hypervisor or operating system should initialize the event log buffer to all zeros and write event log entries to zero after they are processed. If software subsequently observes an all zero event log entry, it should re-read the buffer until a non-zero event log is returned. It is recommended that software detects that the log buffer has not been written by checking for an EventCode (bits 63:60) that is equal to 0000b.

### Fix Planned

No fix planned

## 733 IOMMU PPR Log Ordering Violation

---

### Description

The processor IOMMU does not maintain producer-consumer ordering between the IOMMU peripheral page service request (PPR) log DMA writes and IOMMU MMIO register read completions. The processor core may read stale or uninitialized PPR logs from memory when a read response from the PPR log tail pointer register passes the corresponding PPR log DMA write. A series or burst of PPR log DMA writes would normally be necessary for this ordering violation to be observed.

This erratum only applies in systems where a device is performing Address Translation Service (ATS) requests.

### Potential Effect on System

Software may process a PPR log before it has been completely written, possibly resulting in the IOMMU software not properly processing a page service request. This may result in unpredictable IOMMU behavior.

### Suggested Workaround

The IOMMU driver of the hypervisor or operating system should initialize the PPR log buffer to all zeros and write PPR log entries to zero after they are processed. If software subsequently observes an all zero PPR log entry, it should re-read the buffer until a non-zero PPR log is returned. It is recommended that software detects that the log buffer has not been written by checking for a PPRCode (bits 63:60) that is equal to 0000b.

### Fix Planned

No fix planned

## **734 Processor May Incorrectly Store VMCB Data**

---

### **Description**

Under a highly specific and detailed set of internal timing conditions during a #VMEXIT for a virtual machine guest that has multiple virtual CPUs, the processor may store incorrect data to the virtual machine control block (VMCB) reserved and guest save areas and may also store outside of the VMCB.

### **Potential Effect on System**

Data corruption.

### **Suggested Workaround**

Contact your AMD representative for information on a BIOS update.

### **Fix Planned**

No fix planned

## 737 Processor Does Not Check 128-bit Canonical Address Boundary Case on Logical Address

---

### Description

The processor core may not detect a #GP exception if the processor is in 64-bit mode and the logical address of a 128-bit operation (for example, a octal-word SSE instruction) is canonical on the first byte, but whose final byte crosses over the canonical address boundary. The processor does check the linear address and signals a #GP exception if the linear address is not canonical (for all eight bytes of the operation). Therefore, this erratum can only occur if the segment register is non-zero and causes a wrap in the logical address space only.

In the unlikely event that software causes this wrap, the processor core will execute the 128-bit operation as if the second part of the misaligned access starts at linear address equal to zero.

### Potential Effect on System

None expected, as the normal usage of segment registers and segment limits does not expose this erratum.

### Suggested Workaround

None required.

### Fix Planned

No fix planned

## 739 Processor May Read Branch Status Register With Inconsistent Parity Bit

---

### Description

Under a highly specific and detailed set of internal timing conditions, the processor may read an internal branch status register (BSR) while the register is being updated and may observe a partially written entry with an inconsistent parity bit. When the conditions for this erratum occur, the processor does not actually use the contents of this branch status register, however it may report a parity error machine check exception (#MC).

### Potential Effect on System

The processor reports an uncorrectable machine check exception for a branch status register parity 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]).

### Fix Planned

No fix planned

## 740 Lightweight Profiling May Cause System Hang with Concurrent Stop Clock

---

### Description

The processor may hang if it performs an internal stop-clock event to handle an I/O C-state request or P-state change at approximately the same time that a lightweight profiling (LWP) monitored event overflows its event counter, signalling the need for an LWP event record to be stored. Only LWP record type 2 (instructions retired) or LWP record type 3 (branches retired) events can cause this hang to occur. LWP is enabled once software executes an LLWCP or XRSTOR instruction with a valid LWP control block (LWPCB) address.

### Potential Effect on System

System hang.

### Suggested Workaround

Contact your AMD representative for information on a BIOS update.

### Fix Planned

No fix planned

## 746 IOMMU Logging May Stall Translations

---

### Description

The processor IOMMU L2B miscellaneous clock gating feature may cause the IOMMU to stop processing IOMMU translations due to a perceived lack of credits for writing upstream command processor writebacks, MSI interrupts, peripheral page service request (PPR) logs, or event logs.

### Potential Effect on System

The IOMMU may hang. This may be observed as a system hang.

### Suggested Workaround

BIOS should disable L2B miscellaneous clock gating by setting  
L2\_L2B\_CK\_GATE\_CONTROL[CKGateL2BMiscDisable] (D0F2xF4\_x90[2]) = 1b.

### Fix Planned

No fix planned



## 775 Processor May Present More Than One #DB Exception on REP-INS or REP-OUTS Instructions

---

### Description

When a REP-INS<sub>x</sub> or REP-OUTS<sub>x</sub> instruction is interrupted by a system management interrupt (SMI), the processor does not set RFLAGS.RF to 1b in the SMM save state. After the SMM code executes RSM to return from SMM, any debug exception present on the instruction may get presented again.

### Potential Effect on System

Debug software may observe two or more #DB exceptions for a single execution of REP-INS or REP-OUTS instruction.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## **778 Processor Core Time Stamp Counters May Experience Drift**

### **Description**

Following a long period in a P-state without any core P-state or C-state activity, the time stamp counter for a processor core may appear to drift slightly from other processor cores. This TSC drift does not occur unless the processor has spent over four billion consecutive clocks in a single P-state at C0.

This erratum does not apply if the processor is in the non-boosted software P0 frequency.

### **Potential Effect on System**

System software or software with multiple threads may observe that one thread or processor core provides TSC values that are behind another thread or processor core. This can only happen if the processor core is spending very long intervals in the C0 (running) state and is either pinned to a software P-state lower than P0, or the application power management (APM) behavior of the software running on this core allows the processor to remain in a boosted state without any changes to the P-state.

A single thread operating on a single core can not observe successively stored TSC values that incorrectly decrement.

### **Suggested Workaround**

None required.

### **Fix Planned**

No fix planned

## 786 APIC Timer Periodic Mode is Imprecise

---

### Description

The APIC timer may not properly initialize back to the APIC timer initial count value (APIC380) when it transitions to zero and Timer Local Vector Table Entry[Mode] (APIC320[17]) is configured to run in periodic mode. In this mode, when the APIC timer reaches zero, the next value in the APIC current count register (APIC390) is set to the APIC initial count (APIC380), but the processor may incorrectly add or subtract an offset that is between 0 and 31.

### Potential Effect on System

The standard use of the APIC timer and the level of accuracy required does not make this error significant.

### Suggested Workaround

None.

### Fix Planned

No fix planned

## 787 IOMMU Interrupt May Be Lost

---

### Description

The IOMMU may not send a message-signaled interrupt (MSI) if the interrupt becomes pending around a very small timing window while the software is clearing a previous IOMMU interrupt. Once this erratum occurs, message-signaled interrupts are not sent until the software clears the interrupt pending status.

### Potential Effect on System

The IOMMU driver may not receive IOMMU interrupts. This effect may persist and cause the IOMMU to stop processing transactions since the software normally does not clear the interrupt pending status in the absence of an actual interrupt.

AMD has only observed this erratum with IOMMU version 2 software.

### Suggested Workaround

No workaround is recommended for IOMMU version 1 driver software.

After clearing interrupts, the IOMMU version 2 driver software should read the interrupt status register and process any pending interrupts.

### Fix Planned

No fix planned

## 788 Processor Incorrectly States Support for IOMMU Prefetch Command

---

### Description

The processor does not support the PREFETCH\_IOMMU\_PAGES command, but incorrectly states support for this feature in IOMMU Extended Feature Low[PrefSup] (IOMMU MMIO region, offset 30h bit 0).

### Potential Effect on System

System software may receive an event log (ILLEGAL\_COMMAND\_ERROR) when using a PREFETCH\_IOMMU\_PAGES command. AMD has not observed this erratum with any commercially available software.

### Suggested Workaround

IOMMU software should ignore Extended Feature Low[PrefSup] on affected processors and not use the PREFETCH\_IOMMU\_PAGES command.

### 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 10h-1Fh Processors*, order# 42300
- *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 I/O Virtualization Technology (IOMMU) Specification*, order# 48882
- *Advanced Platform Management Link (APML) Specification*, order# 41918

See the AMD Web site at [www.amd.com](http://www.amd.com) for the latest updates to documents.