Revision Guide for AMD Family 17h Models 00h-0Fh Processors

Publication # **55449** Revision: **1.12**
Issue Date: **June 2018**

Advanced Micro Devices

The information contained herein is for informational purposes only, and is subject to change without notice. While every precaution has been taken in the preparation of this document, it may contain technical inaccuracies, omissions and typographical errors, and AMD is under no obligation to update or otherwise correct this information. Advanced Micro Devices, Inc. makes no representations or warranties with respect to the accuracy or completeness of the contents of this document, and assumes no liability of any kind, including the implied warranties of noninfringement, merchantability or fitness for particular purposes, with respect to the operation or use of AMD hardware, software or other products described herein. No license, including implied or arising by estoppel, to any intellectual property rights is granted by this document. Terms and limitations applicable to the purchase or use of AMD's products are as set forth in a signed agreement between the parties or in AMD's Standard Terms and Conditions of Sale.

Trademarks

List of Figures

List of Tables

Revision History

Date	Revision	Description
June 2018	1.12	Initial public release.

Overview

The purpose of the *Revision Guide for AMD Family 17h Models 00h-0Fh* is to communicate updated product information to designers of computer systems and software developers. This revision guide includes information on the following products:

- AMD Ryzen™ Processor
- 2nd Generation AMD Ryzen™ Processor
- AMD EPYC™ Processor
- AMD Ryzen™ Threadripper™ Processor

Feature support varies by brands and OPNs (Ordering Part Number). To determine the features supported by your processor, contact your customer representative.

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".

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).

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 *Processor Programming Reference (PPR) for AMD Family 17 Model 00h-0Fh Processors*, order# 54945, or the *Open-Source Register Reference for AMD Family 17h Processors*, order# 56255.

Processor Identification

This section shows how to determine the processor revision.

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).

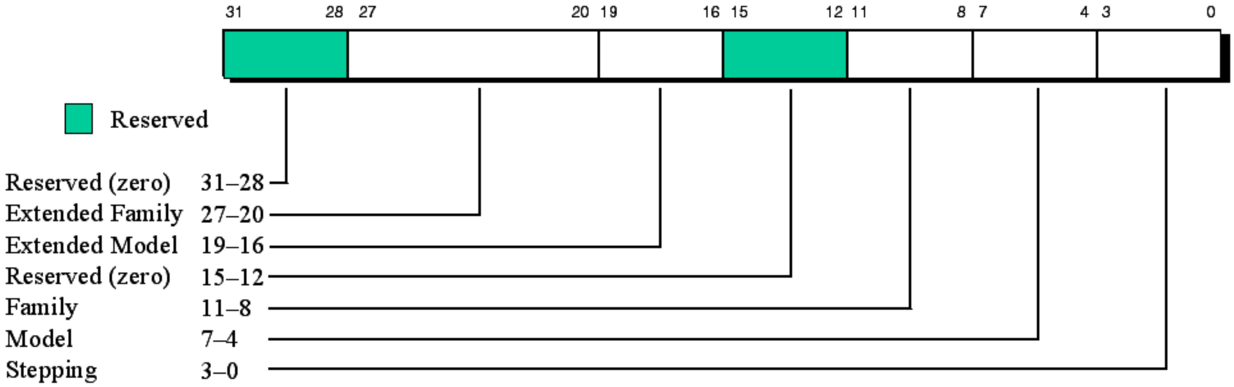

Figure 1. Format of CPUID Fn0000_0001_EAX

The following tables show the identification numbers from CPUID Fn0000_0001_EAX 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 17h Models 00h-0Fh SP3 Processor Revisions

CPUID Fn0000_0001_EAX (Mnemonic)	AMD EPYC™ Processors
00800F12h (ZP-B2)	X

Table 3. CPUID Values for AMD Family 17h Models 00h-0Fh AM4 Processor Revisions

CPUID Fn0000_0001_EAX (Mnemonic)	AMD Ryzen™ Processors	2nd Generation AMD Ryzen™ Processors
00800F11h (ZP-B1)	X	N/A
00800F82h (PiR-B2)	N/A	X

Table 4. CPUID Values for AMD Family 17h Models 00h-0Fh SP3r2 Processor Revisions

CPUID Fn0000_0001_EAX (Mnemonic)	AMD Ryzen™ Threadripper Processors
00800F11h (ZP-B1)	X

Mixed Processor Revision Support

AMD Family 17h processors with different revisions may not be mixed in a multiprocessor system.

Programming and Displaying the Processor Name String

This section, intended for system software 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 system software 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 system software 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 PPR 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 PPR 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].

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.

System software shall program the OSVW_ID_Length to 0005h prior to hand-off to the OS.

Table 5. OSVW ID Length Register

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 system software 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.

Table 6. OSVW Status Register

Bits	Description
63:5	**OsvwStatusBits:** Reserved. OS visible work-around status bits. Read-write.
4	**OsvwId4:** Reserved, must be zero.
3	**OsvwId3:** Reserved, must be zero.
2	**OsvwId2:** Reserved, must be zero.
1	**OsvwId1:** Reserved, must be zero.
0	**OsvwId0:** Reserved, must be zero.

System software shall program the state of the valid status bits as shown in Table 7 prior to hand-off to the OS.

Table 7. Cross Reference of Product Revision to OSVW ID

CPUID Fn0000_0001_EAX (Mnemonic)	MSRC001_0141 Bits
00800F11h (ZP-B1)	0000_0000_0000_0000h

Table 7. Cross Reference of Product Revision to OSVW ID (continued)

CPUID Fn0000_0001_EAX (Mnemonic)	**MSRC001_0141 Bits**
00800F12h (ZP-B2)	0000_0000_0000_0000h
00800F82h (PiR-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. "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 8. Cross-Reference of Processor Revision to Errata

No.	Errata Description	CPUID Fn0000_0001_EAX		
		00800F82h (PiR-B2)	00800F11h (ZP-B1)	00800F12h (ZP-B2)
911	IOMMU Unnecessarily Updates Dirty Bit (D-bit) While Handling Non-supervisor DMA Write Request To Writable Supervisor-only Page	No fix planned		
913	IOMMU Incorrectly Issues Guest Page Table Walk Request as Non-coherent Request	No fix planned		
919	USB tPortConfiguration Timer Incorrectly Resets During Recovery Before LMP (Link Management Packet) Exchange	No fix planned		
923	IOMMU Event Not Logged When Software Programs DTE.HAD Bits Incorrectly	No fix planned		
931	MCA_MISC0[BlkPtr] May Contain Incorrect Value	No fix planned		
937	Unpredictable IOMMU IO_PAGE_FAULT Event Logging For PCIe® Atomic Requests To Protected Pages	No fix planned		
954	Processor Will Shut Down If It Issues a Load That Consumes Poison Data When Error Reporting is Disabled	No fix planned		
955	Processor May Stall If Error Reporting is Disabled and a Cacheable Lock or Table-Walk Load Encounters a Master Abort, Target Abort, or Protection Violation	No fix planned		
965	Incorrect IOMMU IO_PAGE_FAULT Event Logging For Reserved Message Type Interrupt Requests With DTE.IG = 1	No fix planned		
990	Certain Performance Counters For Retire Based Events May Overcount	No fix planned		
1017	FERR (Legacy Floating Point Error) for Thread 0 May be Incorrectly Cleared When Thread 1 Clears Its FERR		X	
1021	Load Operation May Receive Stale Data From Older Store Operation	No fix planned		
1023	Performance Monitor Counter Overflow Interrupts May Fail To Be Delivered When Two or More Counters Are Enabled		X	
1024	Cacheable Load Following Misaligned Cacheable Store Does Not Complete	No fix planned		
1033	A Lock Operation May Cause the System to Hang		X	
1034	Processor May Return Incorrect Faulting Linear Address For a Cacheline-Misaligned Store		X	
1036	When IOMMU Interrupt Remapping Is Enabled the Remapped TM (Trigger Mode) Bit Is Incorrectly Ignored		X	
1037	USB 2.0 Device May Immediately Reconnect After Windows® "Safely Remove Hardware" Procedure	No fix planned		
1038	xHCI Controller May Incorrectly Drop USB 3.0 ISOC Audio Packets		X	
1039	Non-Cacheable Coherent Store May Not Complete If it Follows a Cacheable Access to the Same Cache Line		X	
1042	Processor May Fail To Boot On Systems With Both SPI (Serial Peripheral Interface) and Discrete TPM (Trusted Platform Module) Enabled	No fix planned		
1043	IOMMU May Fail to Deliver an Interrupt or Incorrectly Send an Interrupt to the Host OS	No fix planned		
1044	PCIe® Controller May Hang on Entry Into Either L1.1 or L1.2 Power Management Substate		X	

Table 8. Cross-Reference of Processor Revision to Errata (continued)

No.	Errata Description	CPUID Fn0000_0001_EAX		
		00800F82h (PiR-B2)	00800F11h (ZP-B1)	00800F12h (ZP-B2)
1047	Miss Address Buffer Performance Counter May Be Inaccurate	No fix planned		
1048	Three-Source Operand Floating Point Instructions May Block Another Thread on the Same Core	No fix planned		
1049	FCMOV Instruction May Not Execute Correctly	No fix planned		
1053	When SMAP is Enabled and EFLAGS.AC is Set, the Processor Will Fail to Page Fault on an Implicit Supervisor Access to a User Page	No fix planned		
1054	Instructions Retired Performance Counter May Be Inaccurate	No fix planned		
1057	MWAIT or MWAITX Instructions May Fail to Correctly Exit From the Monitor Event Pending State		X	X
1058	Executing Code in the Page Adjacent to a Canonical Address Boundary May Cause Unpredictable Results		X	X
1059	In Real Mode or Virtual-8086 Mode MWAIT or MWAITX Instructions May Fail to Correctly Exit From the Monitor Event Pending State	No fix planned		
1063	PCIe® Controller Will Generate MSI (Message Signaled Interrupt) With Incorrect Requestor ID	No fix planned		
1067	L3 Performance Event Counter May Be Inaccurate	No fix planned		
1070	16-bit Real Mode Applications May Fail When Virtual Mode Extensions (VME) Are Enabled		X	X
1071	Spurious Level 2 Branch Target Buffer (L2 BTB) Multi-Match Error May Occur	No fix planned		
1076	CPUID Fn8000_0007_EDX[CPB] Incorrectly Returns 0		X	
1080	PCIe® Link Exit to L0 in Gen1 Mode May Incorrectly Trigger NAKs		X	X
1081	Programming MSRC001_0015 [Hardware Configuration] (HWCR)[CpbDis] Does Not Affect All Threads In The Socket		X	X
1083	PCIe® Link in Gen3 Mode May Incorrectly Observe EDB Error and Enter Recovery		X	X
1084	xHCI Host May Fail To Respond to Resume Request From Downstream USB Device Within 1 ms	No fix planned		
1091	4K Address Boundary Crossing Load Operation May Receive Stale Data	X	X	X
1092	USB Device May Not be Enumerated After Device Reset	No fix planned		
1095	Potential Violation of Read Ordering In Lock Operation In SMT (Simultaneous Multithreading) Mode	X	X	X
1096	The GuestInstrBytes Field of the VMCB on a VMEXIT May Incorrectly Return 0h	No fix planned		
1108	MCA Error May Incorrectly Report Overflow Condition	No fix planned		
1109	MWAIT Instruction May Hang a Thread		X	X

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 9. Cross-Reference of Errata to Package Type

Errata	Package		
	AM4	SP3	SP3r2
911	X	X	X
913	X	X	X
919	X	X	X
923	X	X	X
931	X	X	X
937	X	X	X
954	X	X	X
955	X	X	X
965	X	X	X
990	X	X	X
1017	X	X	X
1021	X	X	X
1023	X	X	X
1024	X	X	X
1033	X	X	X
1034	X	X	X
1036	X	X	
1037	X	X	X
1038	X	X	X
1039	X	X	X
1042	X	X	X
1043	X	X	X
1044	X		X
1047	X	X	X
1048	X	X	X
1049	X	X	X
1053	X	X	X
1054	X	X	X
1057	X	X	X
1058	X	X	X

Table 9. Cross-Reference of Errata to Package Type (continued)

Errata	Package		
	AM4	SP3	SP3r2
1059	X	X	X
1063	X	X	X
1067	X	X	X
1070	X	X	X
1071	X	X	X
1076	X	X	X
1080	X	X	X
1081	X	X	X
1083	X	X	X
1084	X	X	X
1091	X	X	X
1092	X	X	X
1095	X	X	X
1096	X	X	X
1108	X	X	X
1109	X	X	X

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 10. Cross-Reference of Errata to Processor Segments

Errata	Processor Segment			
	2nd Generation AMD Ryzen™ Processors	**AMD EPYC™ Processors**	**AMD Ryzen™ Processors**	**AMD Ryzen™ Threadripper™ Processors**
911	X	X	X	X
913	X	X	X	X
919	X	X	X	X
923	X	X	X	X
931	X	X	X	X
937	X	X	X	X
954	X	X	X	X
955	X	X	X	X
965	X	X	X	X
990	X	X	X	X
1017		X	X	X
1021	X	X	X	X
1023			X	X
1024	X	X	X	X
1033			X	X
1034			X	X
1036			X	
1037	X	X	X	X
1038			X	X
1039			X	X
1042	X	X	X	X
1043	X	X	X	X
1044			X	X

Table 10. Cross-Reference of Errata to Processor Segments (continued)

Errata	Processor Segment			
	2nd Generation AMD Ryzen™ Processors	AMD EPYC™ Processors	AMD Ryzen™ Processors	AMD Ryzen™ Threadripper™ Processors
1047	X	X	X	X
1048	X	X	X	X
1049	X	X	X	X
1053	X	X	X	X
1054	X	X	X	X
1057			X	
1058		X	X	X
1059	X	X	X	X
1063	X	X	X	X
1067	X	X	X	X
1070		X	X	X
1071	X	X	X	X
1076			X	X
1080		X	X	X
1081		X	X	X
1083		X	X	X
1084	X	X	X	X
1091		X		X
1092	X	X	X	X
1095		X		X
1096	X	X	X	X
1108	X	X	X	X
1109	X	X	X	X

911 IOMMU Unnecessarily Updates Dirty Bit (D-bit) While Handling Non-supervisor DMA Write Request To Writable Supervisor-only Page

Description

IOMMU incorrectly sets the D-bit in the guest page table when it encounters a DMA write request without supervisor privilege to a writable supervisor-only page.

Potential Effect on System

No functional issue is expected; the non-permitted DMA write request is aborted without any memory content modification. However, the affected pages may be unnecessarily written out to the pagefile by software.

Suggested Workaround

None

Fix Planned

No fix planned

913 IOMMU Incorrectly Issues Guest Page Table Walk Request as Non-coherent Request

Description

When software issues a guest table walk request with DTE.SD=1, IOMMU will issue the table walk request as a non-coherent request just based on the DTE.SD value. It ignores the intermediate guest page PTE.FC value to properly determine if the guest page table walk request should be a coherent request.

Potential Effect on System

A guest page table walk request is issued as non-coherent instead of coherent even when host PTE.FC (Force Coherent) is set.

Suggested Workaround

Software should program DTE.SD=0.

Fix Planned

No fix planned

919 USB tPortConfiguration Timer Incorrectly Resets During Recovery Before LMP (Link Management Packet) Exchange

Description

If the USB link transitions through Recovery before the LMPs are exchanged successfully in U0 state, the tportConfiguration timer erroneously resets during Recovery and restarts once the link enters U0.

Potential Effect on System

None. There is no interoperability or compliance issue because the controller is able to exchange the Port Capability and Port Configuration LMPs after transiting to U0.

Suggested Workaround

None

Fix Planned

No fix planned

923 IOMMU Event Not Logged When Software Programs DTE.HAD Bits Incorrectly

Description

IOMMU fails to log an ILLEGAL_DEV_TABLE_ENTRY event when it encounters software setting of DTE.HAD (Host Access and Dirty Update) bits that is inconsistent with what EFR (Extended Feature Register) specifies. The following are the invalid programming scenarios:

- HASup == 0 & DTE.HAD != 00b.
- HASup == 1 & HDSup == 0 & DTE.HAD == 1xb.
- HASup == 1 & HDSup == 1 & DTE.HAD == 10b.

Potential Effect on System

Unpredictable system behavior when software does not program DTE.HAD correctly.

Suggested Workaround

Software should program DTE.HAD bits according to the AMD I/O Virtualization Technology (IOMMU) Specification, order# 48882, revision 2.63 or later.

Fix Planned

No fix planned

931 MCA_MISC0[BlkPtr] May Contain Incorrect Value

Description

If CPUID_Fn80000007_EBX[ScalableMca] == 1b, the MCA_MISC0[BlkPtr] field is used to indicate the presence of the additional MISC registers. This field is set to 1 regardless of whether additional MISC registers are present.

Potential Effect on System

None expected.

Suggested Workaround

System software must program the MCA_MISC0[BlkPtr] field to 00h in each MCA_MISC0 register except MCA_MISC0_UMC. System software must program the MCA_MISC0[BlkPtr] to 01h in MCA_MISC0_UMC. System software may contain the workaround for this erratum.

Fix Planned

No fix planned

937 Unpredictable IOMMU IO_PAGE_FAULT Event Logging For PCIe® Atomic Requests To Protected Pages

Description

IOMMU has unpredictable IO_PAGE_FAULT event logging when it encounters a PCIe® atomic request accessing protected pages.

The following might occur when IOMMU encounters PCIe atomic requests accessing protected pages;

- when DTE.SA = 1, IO_PAGE_FAULT might be logged when it should be suppressed,
- when DTE.SA = 0, IO_PAGE_FAULT might be suppressed incorrectly.

Potential Effect on System

Unpredictable event logging behavior. PCIe atomic requests to protected pages are aborted as expected.

Suggested Workaround

None.

Fix Planned

No fix planned

954 Processor Will Shut Down If It Issues a Load That Consumes Poison Data When Error Reporting is Disabled

Description

If MCA_CTL_LS[DcDataErr1]==0 and MCG_CTL[LS]==1, the processor will shut down if it issues a load that consumes poison data. This configuration is unsupported except for platform firmware during the boot phase.

Potential Effect on System

System shutdown during boot if an uncorrectable error occurs.

Suggested Workaround

If platform firmware wishes to enable error logging in the MCA, it should program the following registers:

- MCA_CONFIG_LS[McaxEn] to 1b
- MCA_CTL_LS[63:0] to FFFF_FFFF_FFFF_FFFFh
- MCG_CTL[0] to 1b
- CR4.MCE to 1b

These settings will cause a machine check exception to be generated on an error, which platform firmware must handle. Once handled, the platform firmware can continue operation.

Alternatively, if platform firmware does not wish to enable exceptions, platform firmware should program the following register:

- MCG_CTL[0] to 0b

This setting will cause the processor to ignore errors in the load-store unit and will allow the machine to survive poison data consumption. Platform firmware may poll other MCA banks to look for errors that occur during boot (e.g., platform firmware may poll the MCA banks associated with the memory controller to look for DRAM ECC errors).

Before passing control to the operating system, platform firmware should restore the previous state of these registers.

Fix Planned

No fix planned

955 Processor May Stall If Error Reporting is Disabled and a Cacheable Lock or Table-Walk Load Encounters a Master Abort, Target Abort, or Protection Violation

Description

If MCG_CTL[LS]==1, and MCA_CTL_LS[SystemReadDataErrorT0]==0 or MCA_CTL_LS[SystemReadDataErrorT1]==0, the processor may stall if it issues a cacheable lock or table-walk load that encounters a master abort, target abort, or protection violation. The error may also not be logged. This configuration is unsupported except for platform firmware during the boot phase. Because cacheable locks and table-walk loads must be issued to DRAM, these aborts and violations are not expected to occur.

Potential Effect on System

The system may hang or reset.

Suggested Workaround

If platform firmware wishes to enable error logging in the MCA, it should program the following registers:

- MCA_CONFIG_LS[McaxEn] to 1b
- MCA_CTL_LS[63:0] to FFFF_FFFF_FFFF_FFFFh
- MCG_CTL[0] to 1b
- CR4.MCE to 1b

These settings will cause a machine check exception to be generated on an error, which platform firmware must handle. Once handled, the platform firmware can continue operation.

Before passing control to the operating system, platform firmware should restore the previous state of these registers.

Fix Planned

No fix planned

965 Incorrect IOMMU IO_PAGE_FAULT Event Logging For Reserved Message Type Interrupt Requests With DTE.IG = 1

Description

IOMMU will log an IO_PAGE_FAULT event when it encounters any reserved message type interrupt request regardless of DTE.IG setting. Even when DTE.IG = 1, IO_PAGE_FAULT event is logged when it should be suppressed.

Potential Effect on System

Software may encounter unexpected IO_PAGE_FAULT event logging. Reserved message type interrupt requests are aborted as expected.

Suggested Workaround

Software may ignore IO_PAGE_FAULT event log entries for reserved message type interrupt requests from devices with DTE.IG = 1.

Fix Planned

No fix planned

990 Certain Performance Counters For Retire Based Events May Overcount

Description

The processor may experience sampling inaccuracies that may cause the following performance counters to overcount retire-based events when PMCx022[4] is not equal to zero:

- PMCx002 [Retired x87 Floating Point Operations]
- PMCx003 [Retired SSE/AVX Operations]
- PMCx005 [Retired Serializing Ops]
- PMCx0CB [Retired MMX/FP Instructions]

Potential Effect on System

Inaccuracies in performance monitoring software may be experienced.

Suggested Workaround

None

Fix Planned

No fix planned

1017 FERR (Legacy Floating Point Error) for Thread 0 May be Incorrectly Cleared When Thread 1 Clears Its FERR

Description

Under a highly specific and detailed set of internal timing conditions, if thread 0 enters HALT or MWAIT with a pending FERR, then if thread 1 clears its FERR, the FERR for thread 0 may also incorrectly be cleared.

Potential Effect on System

Unpredictable system behavior.

Suggested Workaround

None

Fix Planned

Yes

1021 Load Operation May Receive Stale Data From Older Store Operation

Description

Under a highly specific and detailed set of internal timing conditions, a load operation may incorrectly receive stale data from an older store operation.

Potential Effect on System

Data corruption.

Suggested Workaround

Program MSRC001_1029[13] to 1b. System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1023 Performance Monitor Counter Overflow Interrupts May Fail To Be Delivered When Two or More Counters Are Enabled

Description

Under a highly specific and detailed set of internal timing conditions, when two or more performance monitor counters on the same thread are enabled to generate an interrupt on a counter overflow, they may stop generating interrupts after the first performance monitor counter interrupt event.

Potential Effect on System

Inaccuracies in performance monitoring software may be experienced.

Suggested Workaround

None. Performance monitor counter overflow interrupts will be delivered correctly if only a single performance monitor counter is enabled per thread.

Fix Planned

Yes

1024 Cacheable Load Following Misaligned Cacheable Store Does Not Complete

Description

Under a highly specific and detailed set of internal timing conditions, if a misaligned cacheable store is followed by a cacheable load to a cache line with overlapping address bits [11:0], the load does not complete.

Potential Effect on System

The system may hang or reset.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1033 A Lock Operation May Cause the System to Hang

Description

Under a highly specific and detailed set of internal timing conditions, a Lock operation may cause the system to hang.

Potential Effect on System

The system may hang or reset.

Suggested Workaround

Program MSRC001_1020[4] to 1b. System software may contain the workaround for this erratum.

Fix Planned

Yes

1034 Processor May Return Incorrect Faulting Linear Address For a Cacheline-Misaligned Store

Description

The processor may return incorrect value when reporting the faulting linear address for a cacheline-misaligned store that is not page-misaligned.

- For a non-nested page fault, CR2[11:0] may be incorrect.
- For a nested page fault, EXITINFO2[11:0] of VMCB (Virtual Machine Control Block) may be incorrect.

Potential Effect on System

None, as long as software does not depend on the byte address of faulting cacheline-misaligned store.

Suggested Workaround

If software requires bits [11:0] of the faulting address, then software may decode this information from the faulting X86 instruction and ignore information from EXITINFO2[11:0] or CR2[11:0].

- For non-nested page faults the faulting rIP can be used to read the instruction bytes.
- For nested page faults the Guest Instruction Bytes field in the VMCB provides the instruction to be decoded.

Fix Planned

Yes

1036 When IOMMU Interrupt Remapping Is Enabled the Remapped TM (Trigger Mode) Bit Is Incorrectly Ignored

Description

When IOMMU interrupt remapping is enabled the remapped TM (trigger mode) bit in the IOMMU interrupt remapping table entry is incorrectly ignored, and as a result all remapped interrupts will have TM=0 indicating edge-triggered mode.

Potential Effect on System

Remapped interrupts with TM=1 indicating level-triggered mode will result in IO devices not receiving EOI (end of interrupt).

Suggested Workaround

No workaround. Only use devices with edge-triggered interrupts.

Fix Planned

Yes

1037 USB 2.0 Device May Immediately Reconnect After Windows® "Safely Remove Hardware" Procedure

Description

If Selected Suspend is enabled, a USB 2.0 device may immediately be reconnected after the Windows® "Safely Remove Hardware" procedure completes.

Potential Effect on System

A USB 2.0 device may remain attached to the system after undergoing the Windows® "Safely Remove Hardware" procedure.

Suggested Workaround

To avoid this issue the USB driver should not program the xHCI controller in D3 state when the last connected USB 2.0 device is disabled as a result of the Windows® "Safely Remove Hardware" procedure.

Fix Planned

No fix planned

1038 xHCI Controller May Incorrectly Drop USB 3.0 ISOC Audio Packets

Description

Under a highly specific and detailed set of internal timing conditions, the xHCI controller may incorrectly drop USB 3.0 ISOC audio packets.

Potential Effect on System

Audio stuttering during video playback.

Suggested Workaround

To avoid this issue the USB driver should disable U1 and U2 states for the AMD USB device with vendor ID 0x1022 (USB:ROOT_HUB\VID_1022) and PCI® device ID 0x145C (USB:ROOT_HUB\PID_145C).

Fix Planned

Yes

1039 Non-Cacheable Coherent Store May Not Complete If it Follows a Cacheable Access to the Same Cache Line

Description

Under a highly specific and detailed set of internal timing conditions, a non-cacheable coherent store may not complete if it follows a cacheable access to the same cache line.

Potential Effect on System

The system may hang or reset.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

Yes

1042 Processor May Fail To Boot On Systems With Both SPI (Serial Peripheral Interface) and Discrete TPM (Trusted Platform Module) Enabled

Description

Under a highly specific and detailed set of internal timing conditions, processor may hang while booting on a system when the following conditions occur:

- both SPI (Serial Peripheral Interface) and discrete TPM (Trusted Platform Module) modules enabled, and
- SPI ROM read prefetching is enabled.

Potential Effect on System

System hang while booting.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1043 IOMMU May Fail to Deliver an Interrupt or Incorrectly Send an Interrupt to the Host OS

Description

IOMMU may fail to deliver an interrupt or incorrectly send an interrupt to the host OS under the following conditions:

- The vAPIC (Virtual Advanced Programmable Interrupt Controller) backing page is programmed in IRTE (Interrupt Remapping Table Entry), and
- GA (Guest Virtual APIC) mode is enabled.

Potential Effect on System

Unpredictable system behavior.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1044 PCIe® Controller May Hang on Entry Into Either L1.1 or L1.2 Power Management Substate

Description

Under a highly specific and detailed set of internal timing conditions, the PCIe® controller may hang on entry into either L1.1 or L1.2 power management substate.

This failure occurs when L1 power management substate exit is triggered by a link partner asserting CLKREQ# prior to the completion of the L1 power management stubstates entry protocol.

Potential Effect on System

The system may hang or reset.

Suggested Workaround

Disable L1.1 and L1.2 power management substates. System software may contain the workaround for this erratum.

Fix Planned

Yes

1047 Miss Address Buffer Performance Counter May Be Inaccurate

Description

The processor may experience sampling inaccuracies in the the following performance counter:

- MSRC001_1037[DcMissNoMabAlloc] may not be set.

Potential Effect on System

Performance monitoring software may experience inaccuracies.

Suggested Workaround

None

Fix Planned

No fix planned

1048 Three-Source Operand Floating Point Instructions May Block Another Thread on the Same Core

Description

An uninterrupted stream of three-source operand floating point instructions (e.g. FMA3) on one thread may block floating point instructions from completing on the other thread of the same core. A cache miss or a TLB miss or a branch misprediction would be sufficient to interrupt the stream of three-source operand floating point instructions and prevent the problem.

Potential Effect on System

Unpredictable system behavior.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1049 FCMOV Instruction May Not Execute Correctly

Description

Under a highly specific and detailed set of internal timing conditions, an FCMOV instruction may yield incorrect data if the following sequence of events occurs:

- An FCOMI instruction
- A non-FP instruction that modifies RFLAGS
- An FCMOV instruction

Potential Effect on System

Incorrect results from FCMOV instruction.

Suggested Workaround

Program MSRC001_1028[4] to 1b. System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1053 When SMAP is Enabled and EFLAGS.AC is Set, the Processor Will Fail to Page Fault on an Implicit Supervisor Access to a User Page

Description

When SMAP (Supervisor Mode Access Protection) is enabled and EFLAGS.AC is set to 1b, the processor will fail to page fault on an implicit supervisor access to GDT (Global Descriptor Table), LDT (Local Descriptor Table), IDT (Interrupt Descriptor Table), or TSS (Task State Segment) when they are located in a user page.

Potential Effect on System

If the system data structures GDT, LDT, IDT, or TSS are located in a user page, then an implicit supervisor access may incorrectly gain access to that user page.

Suggested Workaround

None

Fix Planned

No fix planned

1054 Instructions Retired Performance Counter May Be Inaccurate

Description

The processor may experience sampling inaccuracies that may cause the MSRC000_00E9 Read-Only Instructions Retired performance counter to count inaccurately after the processor exits the Core C6 (CC6) state.

Potential Effect on System

Inaccuracies in performance monitoring software may be experienced.

Suggested Workaround

None

Fix Planned

No fix planned

1057 MWAIT or MWAITX Instructions May Fail to Correctly Exit From the Monitor Event Pending State

Description

In the event that the following sequence occurs, a store from another core that matches the MONITOR or MONITORX address range will not cause an exit from the monitor event pending state:

- A thread executes a MONITOR or MONITORX instruction.
- The other thread on the same core changes CR0.CD to 1b. This asserts cache disable for both threads.
- The first thread reads memory to check for desired value.
- The first thread executes an MWAIT or MWAITX instruction.

Potential Effect on System

Unpredictable system behavior.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1058 Executing Code in the Page Adjacent to a Canonical Address Boundary May Cause Unpredictable Results

Description

Under a highly specific and detailed set of internal timing conditions, executing code in the page adjacent to a canonical address boundary may cause unpredictable results.

Potential Effect on System

Unpredictable system behavior.

Suggested Workaround

Supervisor level software (operating systems and hypervisors) should create a guard page between the end of the user-mode accessible virtual address space and the beginning of the non-canonical area to prevent this issue.

Fix Planned

No fix planned

1059 In Real Mode or Virtual-8086 Mode MWAIT or MWAITX Instructions May Fail to Correctly Exit From the Monitor Event Pending State

Description

Under a highly specific and detailed set of internal timing conditions, if a thread executes a MONITOR or MONITORX instruction in real mode or virtual-8086 mode, a store from another core that matches the MONITOR or MONITORX address range may not cause an exit from the monitor event pending state.

Potential Effect on System

Unpredictable system behavior.

Suggested Workaround

None

Fix Planned

No fix planned

1063 PCIe® Controller Will Generate MSI (Message Signaled Interrupt) With Incorrect Requestor ID

Description

The PCIe® controller will generate MSIs with an incorrect Requestor ID of 0x0 on internal interrupt events including:

- Hot-plug
- PME (Power Management Event)
- AER (Advanced Error Reporting)
- DPC (Dynamic Power Control)
- Link Equalization
- Link Bandwidth Notification

Potential Effect on System

Interrupts generated with an invalid Requestor ID may be blocked by the IOMMU if interrupt remapping is enabled; an error log may be generated.

Hot-plugged PCIe® devices will not be configured.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1067 L3 Performance Event Counter May Be Inaccurate

Description

The processor may experience sampling inaccuracies that may cause the L3 Performance Event counter MSRC001_0231 to count inaccurately if

- MSRC001_0230[EventSel] = 0x6, and
- MSRC001_0230[8] (UnitMask[0]) = 1b.

Potential Effect on System

Inaccuracies in performance monitoring software may be experienced.

Suggested Workaround

None

Fix Planned

No fix planned

1070 16-bit Real Mode Applications May Fail When Virtual Mode Extensions (VME) Are Enabled

Description

A 16-bit real mode application may become unresponsive on a system running 32-bit operating system with Virtual Mode Extensions (VME) enabled.

Potential Effect on System

Unpredictable behavior of 16-bit applications on systems running 32-bit operating systems.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1071 Spurious Level 2 Branch Target Buffer (L2 BTB) Multi-Match Error May Occur

Description

Under a highly specific and detailed set of internal timing conditions, the processor may incorrectly log a Level 2 Branch Target Buffer Multi-Match error in MCA_STATUS_IF with extended error code 0xB.

Potential Effect on System

Spurious L2 BTB Multi-Match error may be logged.

If MCA error thresholding is enabled by programming MCA_MISC0_IF[CntEn]=1, the error counter may exceed its threshold and cause the processor to generate threshold overflow interrupts.

Suggested Workaround

System software may contain the workaround for this erratum.

Software should not enable MCA error thresholding in MCA_MISC0_IF[CntEn] to avoid spurious threshold overflow interrupts.

Fix Planned

No fix planned

1076 CPUID Fn8000_0007_EDX[CPB] Incorrectly Returns 0

Description

CPUID Fn8000_0007_EDX[CPB] incorrectly returns 0, indicating the processor does not support Core Performance Boost (CPB). However, the processor does support CPB.

Potential Effect on System

Software may fail to use Core Performance Boost.

Suggested Workaround

Software may ignore CPUID Fn8000_0007_EDX[CPB] and use the Core Performance Boost feature.

Fix Planned

Yes

1080 PCIe® Link Exit to L0 in Gen1 Mode May Incorrectly Trigger NAKs

Description

When the PCIe® link is operating in Gen1 mode and enters electrical idle, the EDB (EnD Bad symbol which marks the end of a nullified Transaction Layer Packet) token is mistakenly forwarded upstream. This unexpected EDB token may incorrectly trigger NAKs (Negative Acknowledgements) when the link exits to L0.

Potential Effect on System

Unexpected NAKs may cause a device operating in PCIe® Gen1 mode to report correctable errors.

Suggested Workaround

If AER (Advanced Error Reporting) is not enabled, no workaround is required.

If AER (Advanced Error Reporting) is enabled, system software may contain the workaround for this erratum.

Fix Planned

No fix planned

1081 Programming MSRC001_0015 [Hardware Configuration] (HWCR)[CpbDis] Does Not Affect All Threads In The Socket

Description

The effect of programming MSRC001_0015 [Hardware Configuration] (HWCR)[CpbDis] is only local to the thread performing the write. The effect is not applied to the other threads in the socket.

Potential Effect on System

Software that assumes MSRC001_0015 [Hardware Configuration] (HWCR)[CpbDis] to have global effect may not have the desired results.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1083 PCIe® Link in Gen3 Mode May Incorrectly Observe EDB Error and Enter Recovery

Description

In Gen 3 mode, the PCIe® Root Port receiver may miss the TLP (Transaction Layer Packet) after a SKP if no IDL is sent before the SKP, causing the port to log a correctable error before the TLP is recovered. This scenario can only happen if the SKP Ordered Set contains 0xC0, causing spurious EDB (EnD Bad symbol) error.

Potential Effect on System

An error-free TLP may be observed as a Bad TLP (correctable error) and cause the link to enter recovery.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1084 xHCI Host May Fail To Respond to Resume Request From Downstream USB Device Within 1 ms

Description

xHCI host may fail to rebroadcast the resume signaling within 1 ms upon receiving resume signal from a downstream hub or device.

Potential Effect on System

Downstream USB hubs or devices requiring a resume signaling response within 1 ms from the xHCI host may malfunction.

Suggested Workaround

For USB hubs or devices that have this requirement in a Microsoft® operating system based system, apply the Microsoft approved "ResetOnResume" USB device registry entry which forces the USB driver stack to reset a device on port resume.

The full description of the registry entry can be found in the Microsoft article at https://msdn.microsoft.com/en-us/library/windows/hardware/jj649944(v=vs.85).aspx.

For example, if the affected USB device has a DeviceInstance value of xxxxyyyyzzzz where

- xxxx is a 4-digit hexadecimal number that identifies the vendor,
- yyyy is a 4-digit hexadecimal number that identifies the product and
- zzzz is a 4-digit hexadecimal number that contains the revision number of the device,

then the registry key for the affected device is as follows:

- [HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\usbflags\xxxxyyyyzzzz]
- "osvc"=hex:00,00
- "ResetOnResume"=hex:00,01

For USB hubs or devices that have this requirement in a Linux® operating system based system, apply the workaround described in the article at https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/drivers/usb/core?h=v4.13-rc5&id=e788787ef4f9c24aafefc480a8da5f92b914e5e6. Customers should contact their Operating System Vendor for availability of this workaround.

Fix Planned

No fix planned

1091 4K Address Boundary Crossing Load Operation May Receive Stale Data

Description

Under a highly specific and detailed set of internal timing conditions, a load operation may incorrectly receive stale data when the following conditions are met:

- there is a preceding store operation to the same address as the load operation, and
- the load operation crosses a 4K address boundary.

Potential Effect on System

In the unlikely event that the condition described above occurs, a load operation would receive stale data that was not updated by the most current write from another logical processor.

Suggested Workaround

Program MSRC001_102D[34] to 1b. System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1092 USB Device May Not be Enumerated After Device Reset

Description

The xHCI controller will ignore the ERDY response from a USB device and timeout incorrectly when the following conditions occur:

- Two or more USB devices are attached either directly to the processor or indirectly via USB hub,
- Reset Device Command is received by any of the USB devices other than the last connected one, causing the USB device to reset and to assume default state with device address 0,
- Software then schedules a new control transfer (e.g. Get_Descriptor command) other than the Set_Address command to the USB device, resulting in a ERDY response from the USB device with address 0.

As a result, the USB device is not enumerated correctly after device reset.

Potential Effect on System

USB device that is not enumerated will not function properly.

Suggested Workaround

Software should always perform Set_Address Command before any other control transfer after Device Reset Command.

Fix Planned

No fix planned

1095 Potential Violation of Read Ordering In Lock Operation In SMT (Simultaneous Multithreading) Mode

Description

Under a highly detailed and specific set of internal timing conditions, a lock operation may not fence a younger load operation correctly when the following conditions are met:

- SMT (Simultaneous Multithreading) is enabled, and
- a lock operation on memory location A, followed by a load operation on memory location B are executing on one thread while
- a lock operation on memory location B, followed by a load operation on memory location A are executing on the second thread on the same core.

This may result in the load operations on both threads incorrectly receiving pre-lock data.

Potential Effect on System

In the unlikely event that the condition described above occurs, a load operation would receive stale data that was not updated by the most current write from another logical processor.

Suggested Workaround

Program MSRC001_1020[57] to 1b. System software may contain the workaround for this erratum.

Fix Planned

No fix planned

1096 The GuestInstrBytes Field of the VMCB on a VMEXIT May Incorrectly Return 0h

Description

On a nested data page fault when CR4.SMAP = 1 and the guest data read generates a SMAP violation, the GuestInstrBytes field of the VMCB on a VMEXIT will incorrectly return 0h instead the correct guest instruction bytes.

Potential Effect on System

A hypervisor will not be able use the GuestInstrBytes field of the VMCB on a VMEXIT to determine what instruction the guest operating system was executing.

Suggested Workaround

To determine what instruction the guest was executing the hypervisor will have to decode the instruction at the instruction pointer.

Fix Planned

No fix planned

1108 MCA Error May Incorrectly Report Overflow Condition

Description

The MSR0000_0001[62] (MCA_STATUS_LS[Overflow]) may be incorrectly set when an MCA error is logged and MSR0000_0001[21:16] (MCA_STATUS_LS[ErrorCodeExt]) is 0x9 (SystemReadDataErrorT0) or 0xa (SystemReadDataErrorT1).

Potential Effect on System

None

Suggested Workaround

None

Fix Planned

No fix planned

1109 MWAIT Instruction May Hang a Thread

Description

Under a highly specific and detailed set of internal timing conditions, the MWAIT instruction may cause a thread to hang in SMT (Simultaneous Multithreading) Mode.

Potential Effect on System

The system may hang or reset.

Suggested Workaround

System software may contain the workaround for this erratum.

Fix Planned

No fix planned

Documentation Support

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

- *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
- *Open-Source Register Reference for AMD Family 17h Processors*, order# 56255
- *Processor Programming Reference (PPR) for AMD Family 17h Model 00h-0Fh Processors*, order# 54945
- *AMD I/O Virtualization Technology(IOMMU) Specification*, order# 48882
- *Open-Source Register Reference for AMD Family 17h Processors*, order# 56255

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