SP5100 Product Errata

Silicon Errata for SP5100

Publication # **46836**	Revision: **3.00**
Issue Date: **June 2012**	

Trademarks

Revision History

Date	Revision	Description
June 2012	3.00	• First public release based on NDA version 3.01

Product Errata Summary

A unique errata reference number (ERN) has been assigned to each erratum within this document for user convenience in tracking the errata within specific revision levels. Table 1 cross-references the revisions of the part to each erratum. An "X" indicates that the erratum applies to the revision. The absence of an "X" indicates that the erratum does not apply to the revision. An "*" indicates advance information that the erratum has been fixed but not yet verified. "No fix planned" indicates that no fix is planned for current or future revisions of the ASIC.

Note: *There may be missing errata numbers. Errata that have been resolved from early revisions of the ASIC have been deleted.*

Table 1: Cross-Reference of Product Revision to Errata

#	Errata Description	ASIC Revision	
		A14	A15
11	Enabling EHCI Dynamic Clock Gating May Cause Bug Code 0xFE System Error	No Fix Planned	
17	USB ISO IN Devices May Not Function Properly	No Fix Planned	
18	System May Not Enter or Resume from S5 After an Unconditional Power Down	X	
19	Non-Posted Writes Using 64-bit Addressing for SKINIT Instructions	X	
20	A-Link Deadlock	X	
21	SMI Re-ordering	X	
22	Transmission Errors on Packet Identifier May Cause USB Host Controller To Reinitialize Device	X	
23	USB Wake on Connect/Disconnect with Low Speed Devices	No Fix Planned	
24	Corrupted Interrupt Vector when both IOAPIC and PIC Controllers Process Interrupts from the Same Source	No Fix Planned	
25	S-state Failures when Message-Triggered C1e is Enabled	No Fix Planned	
26	Excessive Latencies May Cause Overwritten USB OHCI Controller Request	No Fix Planned	
27	Misinterpreted MSI Requests May Result in Corrupted LPC DMA Data	No Fix Planned	
28	Incorrect IOMMU Table Accessed in SATA Combined Mode	No Fix Planned	
30	Nmi_Enable is Altered When Writing to IO_Reg:72h	No Fix Planned	
31	Indeterminate Boot Up State of RTC Bank Selection Bit (DV0)	No Fix Planned	
32	Sleep Resume Hang with STPCLK# Throttling Enabled	No Fix Planned	
33	EHCI Controller State Machine Micro Frame Counter Synchronization	No Fix Planned	
34	Improper Propagation of SATA Message Signaled Interrupts	No Fix Planned	
35	Error in USB Frame List Processing	No Fix Planned	
36	USB Asynchronous Data Cache Error on Back-To-Back DMA	No Fix Planned	
37	USB Babble Detection Logic Disables Unaffected Ports	No Fix Planned	
38	SmiCmdStatus Decoding Failure	No Fix Planned	
40	Non-compliance of the S field of the USB Start-Split Transaction Token	No Fix Planned	
41	Incorrect Implementation of the IOAPIC Delivery Status Bit	No Fix Planned	
42	Incorrect Setting of HPET Num_Tmr_Cap	No Fix Planned	
43	PCI Configuration Trapping May Occur for an Unintended Device	No Fix Planned	

Table 1: (continued) Cross-Reference of Product Revision to Errata

#	Errata Description	ASIC Revision	
		A14	A15
46	LPC SYNC Timeout Violation	No Fix Planned	

Product Errata

11 Enabling EHCI Dynamic Clock Gating May Cause Bug Code 0xFE System Error

Description

A system error has been observed during extended S4 Hibernation or Reboot cycling using the MS PWRTST or other similar utility. The arbiter in the Southbridge that controls the down stream memory traffic to the USB controller does not fully support the EHCI clock gating feature. If the clock gating feature in the EHCI controller is enabled, the arbiter may transfer incorrect memory data to the EHCI controller and cause the controller to not respond back correctly to the USB driver or the device. In such cases, the USB driver may timeout and cause the operating system to report the system error.

Potential Effect on System

The problem may present itself as a system halt with an operating system stop error message with bug check code related to a USB driver failure. The typical operating system error message is BUGCODE_USB_DRIVER bug check value of 0x000000FE. The system error occurs mostly if there are USB devices connected to the system. The failure is intermittent and the failure rate may vary from one system to another. On most systems the failure has been observed to occur after a very large number of reboot cycles (typically more than 1000 cycles). On a small number of systems the issue may be seen within two hundred reboot cycles.

Suggested Workaround

A BIOS workaround is described in section 6.17.1 of the SP5100 Register Programming Requirements document (PID # 44414). The workaround involves disabling the EHCI Dynamic Clock gating Power Management feature in the USB EHCI controller. The feature, when disabled, impacts the total Southbridge power consumption by less than 10 mW.

Fix Planned

No

17 USB ISO IN Devices May Not Function Properly

Description

Data from a USB ISO IN device (e.g., video data from a USB TV tuner) may be corrupted when both the A-Link power savings feature “PLL power down mode” and the L1 link power management features are enabled. With both features enabled, the USB controller may encounter an increased delay in fetching the data from memory. The USB host controller will pre-fetch the data only for ISO type devices to compensate for the delays when fetching the data from the system memory. The increased delay due to the PLL power down mode being enabled (> 5 µS) will prevent the USB controller from getting all the data required from the device in the time allocated by the driver.

Potential Effect on System

To date, this issue has only been observed as corrupted video when testing a USB TV tuner, however, data corruption may also occur with other USB ISO IN devices. The nature of the failures with other USB ISO IN devices will be indicative of the type of the data being transferred.

Suggested Workaround

A USB filter driver will be required that will disable the PLL power down mode if there is a USB 1.1 ISO IN device connected. With this filter driver, the PLL power down mode can be enabled to save additional power when USB 1.1 ISO IN devices are not connected. Customers who are enabling both L1 on AL-LINK and the A-LINK PLL power down mode in the system BIOS will need to include the filter driver as part of their Windows® operating system installation. Systems that do not have the filter driver installed should disable the A-LINK PLL power down mode feature in the system BIOS. As the Linux® operating system USB driver integrates the USB filter driver functionality, the use of a separate filter driver is not required for Linux-based operating systems.

Fix Planned

No

18 System May Not Enter or Resume from S5 After an Unconditional Power Down

Description

Platforms using any PCIe® devices that do not support the PME_Turn_Off broadcast message protocol may experience an intermittent system hang during repeated power cycle testing. Exposure to this failure is contingent on the use of such a PCIe device as well as existence of any one of the following conditions involving an unconditional power down:

a) Cold reset (unconditional power down + wake)

b) Power button 4 second shutdown

c) Thermal shutdown.

d) ASF shutdown command.

Potential Effect on System

This issue will not impact normal shutdown or power up, and is only exposed after repeated reset or power cycling under the scenarios listed above. The failure (a system hang) is typically observed after 100 reset or power cycles.

Suggested Workaround

The system BIOS should be modified to extend the timing of the S5 resume by 3 ms.

Fix Planned

Confirmed fixed in SP5100 Revision A15

19 Non-Posted Writes Using 64-bit Addressing for SKINIT Instructions

Description

Data corruption may occur when 64-bit non-posted write cycles are sent to the Southbridge during the use of SKINIT (Security Kernel Initialization) instructions.

Potential Effect on System

Systems that require the use of SKINIT to support TPM-related security features may not function properly. SKINIT is not used in current shipping operating systems but may be utilized in future operating systems to perform various security initialization functions on systems supporting a TPM (Trusted Platform Module) device. The conditions necessary for this problem to occur are:

1. A TPM device is supported in the system.
2. The operating system / application requires the use of SKINIT instruction for the TPM device.
3. SKINIT is supported by both the chipset and processor.

Suggested Workaround

None.

Fix Planned

Confirmed fixed in SP5100 Revision A15

20 A-Link Deadlock

Description

Under a highly specific and detailed set of stress conditions, including unusually high DMA read and write traffic and host-initiated traffic, a downstream posted or non-posted write may result in a deadlock condition. In order for this to occur, the write must arrive at the Southbridge with a specific pattern of posted writes and responses in the downstream direction as well as non-posted requests and posted writes in the upstream direction.

Potential Effect on System

Under the above unlikely conditions, the system may hang. However, this issue has only been observed when operating in a system extreme-stress test environment (including the use of legacy PCI test cards to generate artificially high DMA bandwidth, a legacy PCI graphics card, and DMA traffic on all other Southbridge ports). The failure has not been observed under normal “real-world” conditions or when the fix to the SMI re-ordering issue (erratum # 21) is disabled.

Suggested Workaround

As there is a very low likelihood of exposure to this issue, it is recommended that no change be done to address this problem. However, the probability of the deadlock can be greatly reduced by disabling the hardware fix to the SMI re-ordering issue (erratum # 21) by setting ABCFG 0x9C[8]=0.

Fix Planned

Confirmed fixed in SP5100 Revision A15

21 SMI Re-ordering

Description

Under a highly specific and detailed set of conditions including unusually high artificially-throttled DMA traffic, the response for an IO write to the SMI command port can pass the upstream SMI, thereby violating an ordering requirement. As a result, SMI interrupt service routines that require the interrupt be taken on the instruction boundary following the IO write to the SMI command port may not function properly.

Potential Effect on System

In extreme cases, and dependent on the SMI BIOS code, system deadlocks and data corruption are possible. However, this issue has only been observed when operating in a system extreme-stress test environment, and has not been observed under normal “real-world” conditions.

Suggested Workaround

A workaround is not required. A hardware fix exists in A12 or later silicon that is enabled when setting ABCFG 0x9C[8]=1. However, as this setting will increase the likelihood of exposure to the A-LINK Deadlock erratum, the BIOS may disable this fix as outlined in the suggested workaround of erratum # 20.

Fix Planned

Confirmed fixed in SP5100 Revision A15

22 Transmission Errors on Packet Identifier May Cause USB Host Controller To Reinitialize Device

Description

When receiving a packet identifier (PID) from a USB device while performing asynchronous data transfers, the USB host controller may not compare the packet type field to its check bits if the incoming packet type decodes as a STALL handshake. If transmission errors on an incoming packet cause a different packet type field in a PID to match the encoding for a STALL handshake, the Southbridge may relay the STALL handshake to the application layer instead of ignoring the packet.

Potential Effect on System

USB host driver software may act on an erroneous STALL handshake and perform a device re-initialization. USB devices should respond to this re-initialization and resume normal operation after a brief delay. If a device is unable to respond correctly to the re-initialization it may disconnect from the host unexpectedly.

Suggested Workaround

None.

Fix Planned

Confirmed fixed in SP5100 Revision A15

23 USB Wake on Connect/Disconnect with Low Speed Devices

Description

Due to an incorrect implementation in the USB logic, the EHCI controller is not able to detect the connection/disconnection of low speed USB 1.1 devices. If the low speed device is not detected, the internal ACPI logic will not be informed that a PME needs to be generated to wake the system when USB device is connected. This issue does not affect low speed devices connected through the OHCI controller.

Potential Effect on System

The system may not wake from the S3 or S4 state when a low speed device is connected to an unused USB port. Similarly, when the system enters the S3 or S4 state with a USB low speed device attached to the port, disconnecting the USB device may not wake the system. Although the connect/disconnect event will not trigger a wake, movement of an attached USB low speed mouse or pressing a key on a USB low speed keyboard will wake the system.

Suggested Workaround

As current SP5100 system BIOS implementations do not enable the “Wake from Connect and Disconnect of USB devices” feature (required to enable the USB PME event and USB resume support), a platform BIOS change is required to enable this option. Section 6.2 of the “AMD SP5100 Register Programming Requirements” document (PID # 44414) outlines the required register settings for enabling this feature. When enabled, this feature will support USB wake on connect/disconnect of high speed and full speed devices, however, a further BIOS workaround is required in order to support low speed devices. Appendix A2 of the same document provides sample code for this required workaround.

Fix Planned

No

24 Corrupted Interrupt Vector when both IOAPIC and PIC Controllers Process Interrupts from the Same Source

Description

Interrupts from the same source initiated from both the IOAPIC and PIC controllers will result in a corrupted interrupt vector.

Potential Effect on System

The manifestation of this issue will be dependent on the hypervisor or operating system and be limited to intermittent error messages that refer to an APIC illegal vector . No functional failures have been observed as a result of the corrupted interrupt vector.

Suggested Workaround

The SBIOS should set SMBUS Cfg 0xAE[6]=1 to resolve the IOAPIC and PIC controller arbitration issue.

Fix Planned

No

25 S-state Failures when Message-Triggered C1e is Enabled

Description

An S-state entry cycle will fail to complete if it was preceded by a message-triggered C1e cycle.

Potential Effect on System

When message-triggered C1e is enabled, a system hang (with no screen display) will occur when the system enters a sleep state. This failure has only been observed when using the S1 sleep state.

Suggested Workaround

In order to support both message-triggered C1e and ACPI S states, a platform BIOS workaround is required that implements an SMI trap that will issue a second sleep command to the PM internal register (PM1 a_CNT).

Fix Planned

No

26 Excessive Latencies May Cause Overwritten USB OHCI Controller Request

Description

Requests from USB OHCI controllers may be overwritten if the latency for any pending request by the USB controller is very long (in the range of milliseconds).

Potential Effect on System

An operating system crash may occur as a result of USB 1.1 devices becoming unresponsive. Although the conditions necessary for exposure to this failure are independent of the operating system and may occur during normal system operation, this failure has only been observed on Microsoft operating systems as a BSOD failure (DRIVER_POWER_STATE_FAILURE STOP 0x0000009F, Sub code 003) during long run ACPI S3/S4 cycling.

Suggested Workaround

A system BIOS update and AMD USB Filter driver version 1.0.14.95 or newer are required in order to avoid this failure. The BIOS change involves disabling OHCI controller pre-fetch on POST, however, given that some USB ISO OUT devices such as USB speakers may experience distorted audio if OHCI controller pre-fetch is disabled, the USB Filter update will enable OHCI controller pre-fetch only when the audio stream to the USB 1.1 ISO out device is active.

A USB filter driver is not supported in the Linux environment, however, Linux kernel version 2.6.32 and newer includes this workaround.

Fix Planned

No

27 Misinterpreted MSI Requests May Result in Corrupted LPC DMA Data

Description

An LPC device that supports DMA may encounter data corruption if used with an operating system that supports HPET MSI (e.g., Windows® 7). This is due to a logic bug in the LPC controller that may cause pending MSI requests to be interpreted as a DMA cycle.

Potential Effect on System

LPC-based components that use DMA such as legacy floppy drives or LPC-based FIR (Far Infrared) devices may fail to operate properly. For example, copying files to a legacy floppy device may result in data corruption in the FAT table of the floppy.

Suggested Workaround

Set SMBUS PCI Cfg 0x43 [7:5] = 000b to disable HPET MSI for platforms that will support LPC devices that use DMA. This change has no affect on Windows Vista® that does not support HPET MSI and will force Windows 7 to use the same legacy type interrupts for HPET as is the case for Windows Vista.

Fix Planned

No

28 Incorrect IOMMU Table Accessed in SATA Combined Mode

Description

Systems with SP5100 SATA set to AHCI or IDE combined mode with SATA port 4 or 5 in use will use the SATA source ID for IDE DMA when consulting the IOMMU tables.

Potential Effect on System

Operating systems where the IDE and SATA tables are separated will experience I/O page faults. Several warnings will be issued during kernel initialization, ultimately leading to the inability to boot the operating system due to the boot disk not being found.

Suggested Workaround

Disable combined mode by setting a platform BIOS callback option to CIMx called "SataIdeCombinedMode" to 0. This change will not have an impact on AHCI/RAID mode, however, in IDE mode, the total number of available ports is reduced by two.

Fix Planned

No

30 Nmi_Enable is Altered When Writing to IO_Reg:72h

Description

A write to IO_Reg:72h (Alternate RTC address) may alter bit 7 (NMI_ENABLE) of IO_Reg:70h. The altered value for IO_Reg:70h[7] is not necessarily related to the value being written into IO_Reg:72h[7].

Potential Effect on System

NMIs may be inadvertently enabled or disabled contrary to the intended error handling intentions and capabilities of the platform.

Suggested Workaround

To ensure that the value of IO_Reg:70h is unaffected after writing to IO_Reg:72h, and to ensure that NMIs cannot occur if inadvertently enabled, the following workaround should be applied on each instance of writing to IO_Reg:72h:

1. Save the current values of IO_Reg:70h and IO_Reg:61h (NMI_STATUS).
2. Write 0Ch to IO_Reg:61h to ensure that NMIs cannot occur if unintentionally enabled.
3. Write the intended value to IO_Reg:72h.
4. Restore IO_Reg:70h to original value.
5. Restore IO_Reg:61h to original value.

Fix Planned

No

31 Indeterminate Boot Up State of RTC Bank Selection Bit (DV0)

Description

The RTC Bank Selection (DV0) bit (RTC_Reg:0A[4]) is not guaranteed to be initialized to the default value (DV0 = 0) by hardware on power cycles involving a VBAT power ramp (i.e., the first power up after the RTC battery is first installed or after the CMOS is cleared via a motherboard jumper). This will result in the software reading data from RTC memory bank 1 instead of bank 0. Of the RTC registers, only the DV0 bit is expected by software to be in the default state (bank 0 selected) on power up.

Potential Effect on System

Unexpected system POST behavior may occur if the DV0 bit comes up in a non-default state (i.e., DV0 = 1). The failure will occur only if the platform BIOS is using the standard bank-dependent indexed register method for accessing the RTC memory through the use of I/O port registers 70h and 71h. A platform BIOS using the AMD proprietary bank-independent indexed register method for accessing the RTC memory (using I/O port registers 72h and 73h) will not be impacted. Once the failure is observed (i.e., DV0 = 1 on power up), the failure will be persistent through warm and cold boot resets. Conversely, if the DV0 bit comes up in the proper default state, it is unaffected by cold or warm resets.

Suggested Workaround

Set DV0=0 during early Southbridge initialization and ahead of any access to RTC RAM. This change is implemented in CIMx version 5.5.0.

Fix Planned

No

32 Sleep Resume Hang with STPCLK# Throttling Enabled

Description

Control bits that enable and disable clock throttling (STPCLK# assertion and de-assertion) reside in registers in the S5 power domain that are not cleared by a reset. If clock throttling is enabled prior to entering the sleep state, a STPCLK# assertion message will be immediately sent upstream to the CPU when the system resumes from the sleep state. The assertion message will cause any read from the BIOS ROM to be blocked from execution as the system has not yet been initialized to allow the STPCLK# message to be sent upstream.

Potential Effect on System

Platforms supporting clock throttling may experience a system hang when resuming from a sleep state if either software or hardware throttling was not disabled prior to entering the sleep state. For software-based clock throttling (enabled and controlled by the operating system if the SBIOS exposes the ACPI capability), the conditions necessary to expose this issue are unlikely under normal operation as the throttling would be disabled by the operating system prior to entering the sleep state. For platforms that support hardware-controlled clock throttling, there is added exposure to this issue as there is no operating system intervention to control the enabling or disabling of the throttling logic.

Suggested Workaround

For platforms that support software-based clock throttling, if conditions exist that preclude the ability of the operating system to disable throttling prior to entering a sleep state, an SMI sleep trap can be implemented to ensure that this occurs. For software-controlled clock throttling, bit 4 of CLKVALUE [CpuControl: 00h] is used to enable or disable throttling.

For platforms that support hardware-based throttling, BIOS code must ensure that throttling is disabled prior to entry into any sleep state as well as for any handling of a shutdown due to a thermal condition. Clearing bit 4 of ThermThrotCntl [PM_Reg:86h) will disable clock throttling. Additionally, bit 0 of ThermThrotCntl can be used to enable a two second delay before clock throttling commences after a shutdown. This will allow sufficient time for the BIOS to initialize the PCIe core, thereby preventing the conditions necessary to expose this issue.

Fix Planned

No

33 EHCI Controller State Machine Micro Frame Counter Synchronization

Description

If, during a narrow window of time within micro frame 7, software enables Periodic Scheduling (USBCMD[4]) with Asynchronous Schedule (USBCMD[5]) disabled and the controller in the RUN state (USBCMD[0] = 1), the EHCI controller's schedule handling logic will lose synchronization with the main state machine and enter a state from which it cannot exit. This scenario has only been observed with Windows 7, however, the conditions necessary to expose this issue may be possible with other operating systems.

Potential Effect on System

For Microsoft-based operating systems, an intermittent BSOD 9F failure may be observed during normal system operation, however, this problem can be exacerbated by stressing the system through sleep cycles while USB flash drives are attached.

Suggested Workaround

If the device driver has both Periodic Scheduling and Asynchronous scheduling disabled, subsequently enabling Periodic Scheduling should use the following sequence:

1. Put the controller in the STOP state (USBCMD[0] = 0)
2. Enable the Periodic schedule
3. Put the controller in the RUN state (USBCMD[0] = 1)

Alternatively, the operating system can remove the conditions necessary to expose this issue by preventing the disabling of Periodic Scheduling. This alternate workaround can be enabled on affected Microsoft operating systems using the "EnHcPeriodicIdle" registry key - refer to Microsoft KB article ID 982091 for details.

Fix Planned

No

34 Improper Propagation of SATA Message Signaled Interrupts

Description

The SATA Message Signaled Interrupt (MSI) from higher port numbers may not properly propagate to port 0 before being sent to the driver.

Potential Effect on System

On a platform with SATA MSI enabled and activity occurring on SATA devices connected to three or more ports, if the failing condition occurs, the driver will not receive the interrupt and the system will hang.

Suggested Workaround

Do not expose the MSI capability in the SATA controller. This change is implemented in CIMx version 6.0.0.

Fix Planned

No

35 Error in USB Frame List Processing

Description

An error in processing the USB frame list may result in the USB controller reading from memory address 0 and, if the contents in that location correspond to a valid Transfer Descriptor, writing to a memory address location outside of the EHCI controller's memory space. The necessary conditions to expose this issue are systems that encounter longer than normal latency (i.e., exceeding 125 µs) on DMA reads and software that implements a USB Periodic Frame List Structure where the Frame List Link Queue Head pointer is a null pointer with T-bit = 1.

Linux, VMware and any custom (in house) BIOS implementations that implement this particular USB Periodic Frame List Structure are exposed. There is no exposure to Microsoft Windows operating systems.

Potential Effect on System

The manifestation of this problem is dependent on the function of the memory address mistakenly accessed by the USB controller. Observed cases lead to system hangs due to a corrupted interrupt vector.

Suggested Workaround

USB Periodic Frame List Structures must be generated where the periodic frame list element pointer sets T-bit = 0 and uses a pointer to a valid but inactive queue head. This change is included in USB driver updates to VMware (ESX 4.1P3 and ESX 5.0) and Linux (version 2.6.37 kernel).

Fix Planned

No

36 USB Asynchronous Data Cache Error on Back-To-Back DMA

Description

An incorrect implementation of the USB data cache address logic may result in data being cached with an incorrect address during a DMA read access that crosses a 4KB boundary. The data that should be cached using addresses in the second page is incorrectly cached with an address corresponding to the same offset in the first page. No issue exists if this address is not used within 800 microseconds when the cache is purged, however, in the event that another USB operation (read or write) corresponds to this incorrect cached address, the EHCI controller may hang (if this subsequent operation is a write), or it may use incorrect data (if this subsequent operation is a read). This erratum is only exposed with USB 2.0 devices connected to ports of the EHCI controller.

Potential Effect on System

This issue has only been observed as a hang of the EHCI controller (affecting all devices attached to that controller) when using a particular USB LAN adapter.

Suggested Workaround

An optional workaround of disabling the USB asynchronous data cache is available in order to remove exposure to this issue. This is achieved via a platform BIOS callback option to CIMx called EhciDataCacheDis introduced in CIMx version 6.2.0. Setting EhciDataCacheDis = 1 will disable the cache, however, the default in CIMx is cache enabled (EhciDataCacheDis = 0).

Fix Planned

No

37 USB Babble Detection Logic Disables Unaffected Ports

Description

USB logic that is used to detect and handle babble (any unexpected USB bus activity that persists beyond a specified point in a micro frame) may disable all ports of the EHCI controller.

Potential Effect on System

This issue has only been observed during USB bus enumeration of a particular USB flash drive under Windows 7, Windows Vista and Windows Server 2008. Failures have not been observed in the Linux environment. Any event that would involve enumeration of this device (e.g., hot plugging or resuming from S3 or S4) would expose the platform to the condition where devices on ports of the same EHCI controller are disabled. The affect on other USB devices attached to the same EHCI controller is dependent on whether the devices were active when the babbling condition started. Active devices may get disabled and will not be successfully re-enumerated by the operating system's USB hub driver unless they are disconnected and re-attached. Inactive devices may get disabled, however, these will be promptly re-enabled by the USB hub driver.

Suggested Workaround

Disable babble detection by setting EOR Misc Control (EOR_Reg : EHCI_EOR + 9Ch[11]) = 1b for each EHCI controller. With this workaround in place, the babbling device will be stopped and re-enumerated by the operating system due to the unexpected activity on the USB bus, however, other non-offending devices on the same EHCI controller will be unaffected.

Fix Planned

No

38 SmiCmdStatus Decoding Failure

Description

The SMI Command Port Status register (SmiCmdStatus [SmiCmdBlk: 01h]) is not decoded properly when the IMC is simultaneously accessing ACPI-related register blocks. As a result, write cycles to SmiCmdStatus (located at SMI Command Port Base + 1 byte) will complete, but the register is not updated. Similarly, reads to SmiCmdStatus may not return the actual register contents. On IMC-enabled platforms where SmiCmdStatus is used, the improper decoding of this register does not affect SMI generation or IMC reads or writes.

Potential Effect on System

Some platform BIOS code bases use SmiCmdStatus as a status register during SMI processing. For these platforms, failure symptoms may vary depending on the specific usage of this register. The observed failure was a system soft hang during POST after resuming from the S4 state.

Suggested Workaround

For IMC-enabled platforms that make use of SmiCmdStatus, read or write to the status register using 16-bit accesses to SmiCmdPort [SmiCmdBlk: 00h]. Additional details of this workaround (to ensure that SMI generation is not affected) are available in section 14.1 of the AMD SP5100 BIOS Developer's Guide (PID # 44415, version 3.01 and later).

Fix Planned

No

40 Non-compliance of the S field of the USB Start-Split Transaction Token

Description

The EHCI controller does not fully comply with the definition of the speed field (S bit) of the Start-Split Transaction Token as defined in the USB 2.0 specification. The requirement that the S bit must be set to 0 for isochronous IN start/splits is not met.

Potential Effect on System

Isochronous devices attached to USB hubs that check the S bit in both start-split and end-split transactions will not function. This issue has only been observed using a USB 2.0 Full Speed camera attached to one specific vendor hub device.

Suggested Workaround

There is no workaround for this issue. Due to this erratum, the EHCI controller does not support hubs that check the speed bit for both start split and end split transactions.

Fix Planned

No

41 Incorrect Implementation of the IOAPIC Delivery Status Bit

Description

The implementation of the APIC hardware does not comply with the functionality of the Delivery Status Bit (DSB) as defined in the IOAPIC specification.

Potential Effect on System

Diagnostics that check the functionality of the DSB may report an error. No functional failures have been observed during normal system operation outside of such diagnostic environments.

Suggested Workaround

None required. Software normally uses other means of determining interrupt status and does not rely on the DSB.

Fix Planned

No

42 Incorrect Setting of HPET Num_Tmr_Cap

Description

The Num_Tmr_Cap field (bits [12:8] of the HPET General Capabilities and ID Register located at the base address defined at SMBUS PCI_Reg: B4h) is incorrectly set to 03h, indicating that four HPET timers are implemented. Only three HPET timers are implemented.

Potential Effect on System

AMD is not aware of any production software that uses more than two HPET timers. However, software that relies on the value of this register and attempts to access more than three HPET timers may experience unexpected behavior.

Suggested Workaround

Software should not use the returned value of the Num_Tmr_Cap field of the HPET General Capabilities and ID Register and instead assume that it is set to 02h to correctly reflect that three HPET timers are implemented.

Fix Planned

No

43 PCI Configuration Trapping May Occur for an Unintended Device

Description

The southbridge may not trap PCI configuration cycles for the intended device if the trap is enabled for a PCI device number that is greater than 0xC. The southbridge may instead trap a cycle for another PCI device if its device number is greater than 0xC and if the PCI configuration address bits match what is being targeted in the trap. The two available PCI configuration traps are enabled by programming registers ProgramPciConfigIndex (PM_Reg: AAh) and ProgramPciConfigBus (PM_Reg: ABh).

Potential Effect on System

Software may receive unexpected traps, resulting in unexpected behavior if traps are configured for device numbers greater than 0xC.

Suggested Workaround

PCI configuration trapping should only be used when the device number is less than or equal to 0xC.

Fix Planned

No

46 LPC SYNC Timeout Violation

Description

The LPC host controller does not meet the SYNC timeout requirements as defined in the LPC specification where a memory, IO or DMA cycle started by the host can be aborted if it observes three consecutive clocks without a defined SYNC. Instead, the LPC controller will abort the cycle if it observes two clocks without a defined SYNC. LPC devices that do not respond within two clocks will consistently fail to claim any cycles.

Potential Effect on System

The observed failure was limited to an LPC-to-ISA bridge where ISA devices behind the bridge did not get detected by the operating system. To ensure that other LPC devices have an opportunity to claim the cycle first, this LPC-to-ISA bridge device intentionally waits for the third clock before claiming and passing the memory, IO or DMA cycle to the ISA bus. No failures have ever been observed on commonly used LPC devices such as SIOs, embedded controllers, BMCs or LPC ROMs.

Suggested Workaround

None

Fix Planned

No