AMD Geode™ SC1200/SC1201 Processor Silicon Revision D3.3 Specification Update

Scope

This document discusses known issues of silicon revision D3.3 of the AMD Geode™ SC1200/SC1201 processor. The table below provides a summary of the issues. A detailed description of each issue, its impact and a recommended resolution/fix follow.

Silicon revision D3.3 is denoted as “33” in the part marking on the package (i.e., “(M) (C) 1999 33”).

The D3.3 silicon is only available in the BGU481 (481 terminal Ball Grid Array Cavity Up) Pb-free (lead-free) package (aka TEPBGA package).

Software Readable IDs for Rev D3.3

- GX1 Processor Module, Configuration Register, DIR1 Offset FFh = 81h.
- General Configuration Block, Revision Register, Offset 3Ch:
  — SC1200 = 04h.
  — SC1201= 05h.
- General Configuration Block, Revision Register, Offset 3Dh = 07h.
- Video Processor Module, F4BAR0+Memory Offset 48h:
  — Revision ID, bits [15:8] = 02h.
  — Device ID, bits [7:0] = 57h.

Issues Summary

Issue #[1]	Description
Section 1.0 "Configuration"	
1.4	IDE/TFT signals float after reset
1.5	Pulse on IDE/TFT signals, Parallel Port/TFT signals and FP_VDD_ON during system power-up
1.9	Many signals may glitch during system power-up
Section 2.0 "GX1 Processor Module"	
2.1	Incorrect CURRENT_IP field in SMI header
2.2	RSM (Resume from SMM) truncates page-granular CS (Code Segment) limit
2.3	SDRAM CAS latency of 1 not supported
2.4	Memory writes in SMI handler may have A20 in their address cleared
2.5	Double fault handled as general protection fault
2.6	Call ESP does not work
2.7	PCI LOCK# signal ignored
2.8	PCI REQs must not go active during reset
2.9	Call at beginning of Code Segment causes general protection fault
2.10	Self-modifying code can cause a page fault (PF)
2.11	Time Stamp Counter stops during Suspend
2.13	WORD access to Ports 23 and 24 sends half of access off-chip
2.14	Reset values for certain Memory Controller registers undefined
2.15	Writes to the DC_TIMING_CFG register sometimes fail
2.16	Different behavior of EFLAGS during INTR handling
2.18	Time Stamp Counter rollover

Issues Summary (Continued)

Issue #[1]	Description
2.19	Outbound bursts on FPCI may fail
2.21	DC high priority signal can get stuck if the FIFO underruns
Section 3.0 "Video Processor Module"	
3.10	Direct VIP (DVIP) does not work with TFT or CRT displays
3.11	Video Processor Display Mode register may not be written when video is not enabled in the Display Controller
3.12	Interference while accessing Video Processor Palette Data register during vertical blank
3.15	Video buffer size in Display Controller must be bigger than Video Processor's window size
3.17	Chroma key use causes video window errors
3.20	Alpha value 255 with chroma key comparison results in video only instead of graphics only
3.21	The left 1-pixel column of an alpha window is graphics only (and not blended) on any left boundary with an alpha window with alpha value 255
3.22	Loading alpha and alpha increment values sometimes fail
3.23	VOP Odd/Even field Indicator bit is incorrect
3.24	YUV to RGB Color Space Converter is inaccurate
3.25	First and/or last line of a video window may flicker with flicker filter enabled
Section 4.0 "Core Logic Module"	
4.1	GPIO status bits not always cleared
4.2	IDE bus master buffer address must be DWORD aligned
4.3	ACPI register access incorrectly disabled
4.4	Simultaneously use of CPU throttling and Sleep state does not work
4.12	PCI burst read with wait states may hang
4.13	PCI special cycles not replicated from internal PCI to external PCI
4.17	Access crossing chip select boundary may affect IOCS0# and IOCS1# address mapping
4.20	PCI/ISA multiplexed signals are left in an unknown state
4.22	Bandwidth limitation on PCI bus during Capture VIP
4.23	In DVIP mode, when the input signal is weak or not present, both video and graphics are not stable
4.24	User Defined Traps when pointed to IDE do not work
4.25	Can not boot from LPC bus using some LPC ROM
4.26	Clearing bit 0 in the IDE Bus Master Command register does not stop the current UDMA transfer
4.27	An IDE bus master transfer of exactly 64 KB cannot be done
4.29	PCI deadlock condition with IDE bus master
4.31	PCI compliance not met on PCI bus
4.32	IDE bus master transfers of less than 20h are not supported
4.33	PCI address stepping masters not supported on GNT1#
4.35	Reprogramming PIT counter mode does not reset counter latches
4.36	Changing Flash data width after boot does not work

Issues Summary (Continued)

Issue #[1]	Description
4.38	ACPI C3 does not wake by interrupt
4.39	Debounce circuit does not work
4.40	Left/right audio output swapped
4.41	IDE reset output misbehaves during software initiated reset
Section 6.0 "Chip I/O"	
6.6	Limited input voltage

1. Issue numbers may not be sequential since issues are omitted once they are resolved.

Issues

1.0 Configuration

1.4 IDE/TFT signals float after reset

Description: The IDE/TFT multiplexed signals float after reset. TFT displays require that all interface signals be at ground until after the TFT panel sequence has started.

Implications: If a system is designed that uses these signals as the TFT interface, the “signals at ground” requirement will not be met and the TFT panel could be damaged.

Resolution: Pull down each of the affected signals with a 2.5 KΩ resistor.

1.5 Pulse on IDE/TFT signals, Parallel Port/TFT signals and FP_VDD_ON during system power-up

Description: The IDE/TFT multiplexed signals, Parallel Port/IDE multiplexed signals, and FP_VDD_ON signal, have approximately a 5 ms pulse on them at power-up. TFT displays require that all interface signals be at ground until after the TFT panel sequence has started.

Implications: If a system is designed that uses these signals as the TFT interface, the “signals at ground” requirement will not be met and the TFT panel could be damaged.

Resolution: Apply V_{CORE} and V_{IO} simultaneously, together with the supply voltage of other system components that may drive the SC1200/SC1201 chip, or external hardware must be used to isolate the pulse on the affected signals so that it does not reach the TFT panel. This can be done with quick switches, buffers, gates or other means.

1.9 Many signals may glitch during system power-up

Description: Many signals may produce a glitch of up to approximately 5 ms at system power-up. This glitch occurs due to power being applied to V_{IO} with V_{CORE} not yet applied. Under this condition, the output buffers may have undefined output control from the core logic which produces the glitch. This glitch has been observed on POR#, PCIRST#, GPIOs and on other signals (see issue 1.5).

Implications: Any external circuitry could misbehave due to this glitch. For example, external circuitry that uses POR# might not reset properly.

Resolution: Apply V_{CORE} before V_{IO} or simultaneously with V_{IO}, and apply V_{CORE} before any other supply voltages that may backdrive into the chip. Or, use external hardware to isolate these glitches from reaching external devices that are sensitive to the glitches. This can be done with a quick switch, buffer, gate, or other means.

2.0 GX1 Processor Module

2.1 Incorrect CURRENT_IP field in SMI header

Description: When two SMIs overlap, the CURRENT_IP field of the SMI header for the second SMI wrongly contains EFLAGS instead of CURRENT_IP.

Implications: None. The CURRENT_IP field is not normally used in SMM code, so this has typically not been a problem.

Resolution: If required, code is available that allows the SMM handler to calculate the CURRENT_IP field.

2.2 RSM (Resume from SMM) truncates page-granular CS (Code Segment) limit

Description: When RSM loads the CS segment limit from the SMI header, it truncates it to 20 bits. If the CS segment was page-granular, it shifts left 12 bits and loses the upper 12 bits of the original limit.

Implications: The system executes code at the wrong location after an RSM to a page-granular CS segment.

Resolution: A software workaround for this issue is implemented in the processor’s SMI handlers. In the middle of the handler, PAGE_GRAN (bit 31 of the CS segment field) is tested. If set, the limit field in the SMI header is shifted right by 12 bits. If not set, nothing is done.

2.3 SDRAM CAS latency of 1 not supported

Description: When CAS latency is set to 1, the Memory Controller does not pick up read data properly.

Implications: CAS latency of 1 cannot be used.

Resolution: None. The impact of this is minor, as there are few (if any) SDRAMs that support this setting.

2.4 Memory writes in SMI handler may have A20 in their address cleared

Description: If a memory write cycle occurs that has A20 set near an RSM instruction, the write may be posted and delayed. When the write cycle is actually executed, the Force A20 logic is applied.

Implications: This can route the data to the wrong address. Unpredictable system behavior can result.

Resolution: Avoid memory write cycles that have A20 set near the RSM, or execute an I/O cycle before the RSM which forces any posted write to execute before the RSM executes.

2.5 Double fault handled as general protection fault

Description: A pending CLI causes a CPU privilege level exception. The trap gate points to a “not present” code segment. Both of these faults are contributory class exceptions.

Implications: A double fault should result from this situation. However, the “not present” fault is taken instead. This is not a functional issue as this fault condition is a result of faulty code and a fault is taken (although the incorrect fault).

Resolution: None required.

2.6 Call ESP does not work

Description: Call ESP instruction is broken.

Implications: This instruction exists because of how the call register instruction is created. This instruction is never used because using this call, and managing the stack, become extremely difficult if not impossible.

Resolution: None required. Do not use this instruction.

2.7 PCI LOCK# signal ignored

Description: The processor ignores the LOCK# signal when PCI bus masters assert LOCK# during a bus transaction.

Implications: This breaks PCI compliance.

Resolution: None.

2.8 PCI REQs must not go active during reset

Description: If a PCI REQ# goes active during reset, the processor’s arbiter may not function correctly after reset goes inactive. The VGA frame buffer is broken.

Implications: None, as long as PCI REQ# is pulled up during reset.

Resolution: None.

2.9 Call at beginning of Code Segment causes general protection fault

Description: A segment exists with a base address that is not 16-byte aligned, and the limit of that segment is at the maximum (FFFFFFFFh). A call instruction is made to the beginning of the segment, which happens to fall in the middle of the 16-byte line fetch. The limit checking hardware is unable to discern that the actual code execution has crossed the limit. It thus assumes that the limit has been crossed, because the line fetch contains both the beginning and the end of the segment, and causes a general protection fault.

Implications: This is a real coding hazard. However, coding practices are such that when a maximum segment is created the base is zero (which is 16-byte aligned).

Resolution: None.

2.10 Self-modifying code can cause a page fault (PF)

Description: A memory write is generated due to an STOS instruction on a page boundary which modifies the STOS instruction. This is followed by a JCC instruction that returns the IP back to the former location of the STOS instruction. The refetch then occurs, but to the wrong address.

Implications: Self-modifying code that executes as described fails.

Resolution: None.

2.11 Time Stamp Counter stops during Suspend

Description: When the processor is in Suspend due to SUSP#/SUSPA# or in HALT with the SUSP_HLT bit (CCR2, Index C2h[3]) set, the Time Stamp Counter stops.

Implications: This is different from other CPUs.

Resolution: None.

2.13 WORD access to Ports 23 and 24 sends half of access off-chip

Description: Executing a WORD I/O cycle to Port 23 is a misaligned cycle that the processor converts into two BYTE cycles. When MAPEN = 1 (CCR3, Index C3h[4]), one cycle goes to Port 23 and the other to Port 24. The Port 23 access does not go off-chip, since that is a CPU I/O port, however, the Port 24 cycle goes off-chip.

Implications: None. There is typically nothing at Port 24.

Resolution: Access Port 23 using BYTE wide I/O instructions.

2.14 Reset values for certain Memory Controller registers undefined

Description: Reset values for the following Memory Controller registers are undefined and should be:
MC_MEM_CTRL1 = 248C0040h
MC_MEM_CTRL2 = 00000801h
MC_BANK_CFG = 41104110h
MC_SYNC_TIM1 = 2A733225h

Implications: If these registers are not written before operation, the Memory Controller may function incorrectly.

Resolution: Before configuring the Memory Controller, use software to set these registers to the reset values above.

2.15 Writes to the DC_TIMING_CFG register sometimes fail

Description: First writes to this register fail. Subsequent rewrites usually succeed if a sufficient delay occurs before the next read/write to a GX1 register.

Implications: The Display Controller may not function as expected.

Resolution: Ensure that writes to the DC_TIMING_CFG register are followed by a delay of at least 200 µs.

2.16 Different behavior of EFLAGS during INTR handling

Description: If an IRQ occurs during EFLAGS style, CPU ID support detection, bit 21 is cleared. This is different from an Intel CPU.

Implications: Possible compatibility problems.

Resolution: Disable IRQs during manipulation of upper bits in EFLAGS.

2.18 Time Stamp Counter rollover

Description: The upper 32 bits of the Time Stamp Counter (TSC) increment three core clocks before the lower 32 bits rollover. If the TSC is read and EAX is FFFFFFFDh, FFFFFFFEh, or FFFFFFFFh, then EDX will have incremented.

Implications: The TSC cannot be read reliably.

Resolution: (1) Use the TSC as a 32-bit counter. (2) When the TSC is read and EAX equals FFFFFFFD, FFFFFFFEh, or FFFFFFFFh; then decrement the EDX value by 1.

2.19 Outbound bursts on FPCI may fail

Description: Outbound bursts from the GX1 module to the FPCI bus sometimes fail.

Implications: Only one scenario has been discovered in which outbound bursts can occur from the GX1 module. If a memory write occurs that causes a cache eviction to the memory region controlled by BC_XMAP and the region is defined as cacheable and write protected, the memory cycle caused by the eviction will burst on the FPCI bus. Unexpected behavior will result, often leading to a system crash.

Resolution: None. Either do not cache and write protect regions controlled by BC_XMAP or make sure memory writes never occur to those regions.

2.21 DC high priority signal can get stuck if the FIFO underruns

Description: The Display Controller’s high priority signal to the Memory Controller is intended to lock out all non-refresh memory requests to prevent FIFO underruns. This high priority signal is activated if the display FIFO level drops below a programmed value (Display FIFO High Priority Start Level in DC_GENERAL_CFG, GX_BASE+8304h [11:8]). However, if the FIFO does underrun, the high priority signal is stuck active until the end of the frame is displayed. The CPU cannot access the Memory Controller during that time.

Implications: If the display FIFO underruns, the CPU is not able to access system memory until the end of the display frame causing unpredictable delays in CPU execution. Interrupts handling latencies can increase and system instabilities can result.

Resolution: Make sure there is sufficient system memory bandwidth available that prevents the display FIFO from underrunning:

– Operate memory subsystem at maximum supported speed.

– If possible, reduce resolution and/or refresh rate of the display.

– If possible, reduce color depth of the graphics.

3.0 Video Processor Module

3.10 Direct VIP and GenLock may not be compatible with TFT/CRT display

Description: DVIP can not be guaranteed operational on CRT/TFT displays due to the GenLock mechanism's slight modification of the sync signals. Many CRT/TFT displays will not correctly operate with modified sync signals.

Implications: AMD will not guarantee DVIP operation on CRT/TFT displays in general. However, with some displays DVIP may operate correctly.

Resolution: None. Use Capture VIP with TFT and CRT displays.

3.11 Video Processor Display Mode register may not be written when video is not enabled in Display Controller

Description: When video is not enabled in the GX1 module's Display Controller, VIDE = 0 (GX_BASE+ Memory Offset 8304[28]), the Display Mode register (F4BAR0+Memory Offset 400h) may not be written.

Resolution: Set VIDE = 1 prior to writing the Display Mode register.

3.12 Interference while accessing Video Processor Palette Data register during Vertical blank

Description: The Palette Data register (F4BAR0+Memory Offset 20h) might not be written correctly if accessed during the vertical blank. This register provides the video palette data. This data can be read or written to the gamma correction RAM (palette) via this register. Prior to accessing this register, an appropriate address should be loaded to the Palette Address register (F4BAR0+Memory Offset 1Ch). Subsequent accesses to the Palette Data register cause the internal address counter to be incremented for the next cycle.

Resolution: Wait for active period before writing to the Palette Data register by reading the Vertical Not Active status bit in the GX1 module's Display Controller (GX_BASE+Memory Offset 8308h[30]).

3.15 Video buffer size in Display Controller must be bigger than Video Processor's window size

Description: Programming the video buffer size in the Display Controller (GX_BASE+Memory Offset 8328h[29:16]) with the same size as the Video Processor's window (F4BAR0+Memory Offset 08h, and 0Ch) causes the last lines in the window to be duplicated.

Implications: None.

Resolution: Program the Display Controller's video buffer size with a value that includes one more line than the height of the Video Processor's window.

3.17 Chroma key use causes video window errors

Description: When video data is compared to chroma key (F4BAR0+Memory Offset 04h[20] = 1) in YUV blending mode, the video window begins 4 pixels to the left of the requested horizontal position, and a wraparound of 4 pixels occurs on the left of the video window.

Implications: Chroma key can not be used with video data when in YUV blending mode.

Resolution: None

3.20 Alpha value 255 with chroma key comparison results in video only instead of graphics only

Description: When performing alpha blending with alpha value 255 using chroma key comparison, only video is seen instead of graphics in the alpha window.

Implications: An alpha value of 255 cannot be used with chroma key comparison.

Resolution: When chroma key comparison is used with alpha blending, use an alpha value of 254 where 255 is required. When chroma key comparison is used with the alpha auto-increment feature, monitor the Video Processor's Alpha Watch register (F4BAR0 +Memory Offset 94h); when its value reaches 254, disable auto-increment and load an alpha value of 254. Using an alpha value of 254 instead of 255 may result in a slight video component inside the alpha window (instead of graphics only) depending on the graphics and video colors.

3.21 The left 1-pixel column of an alpha window is graphics only (and not blended) on any left boundary with an alpha window with alpha value 255

Description: On the boundary between two alpha windows, if the alpha value of the window left of the boundary is 255, then the left 1-pixel column of the other window will behave as alpha value 255 (graphics only) as well. This only happens if the window with alpha value 255 has lower priority than the second window.

Implications: One column of graphics (instead of a blended image) might appear in the boundary between two such alpha windows.

Resolution: In situations which match the above description, use an alpha value of 254 where 255 is required. When activating alpha auto-increment as well, monitor the Video Processor's Alpha Watch register (F4BAR0+Memory Offset 94h); when its value reaches 254, disable auto-increment and load an alpha value of 254. Using an alpha value of 254 instead of 255 may result in a slight video component inside the alpha window (instead of graphics only) depending on the graphics and video colors.

3.22 Loading alpha and alpha increment values sometimes fail

Description: When the command that writes the alpha value (ALPHAx_VAL) and increment value (ALPHAx_INCR) to the alpha registers (Alpha1, Alpha2, and Alpha3, F4BAR0+Memory Offset 6Ch, 7Ch, and 8Ch, respectively) is performed during the vertical sync, the operation always fails. The operation also fails at other specific times during the vertical active. However, the failure rate is about 0.5% when the operation is done during the active video period.

Implications: The alpha and increment values can not be reliably written.

Resolution: Load the alpha and increment values. Then check in the mirror register (F4BAR0+Memory Offset 94h) that the new value was loaded. If the new value was not loaded successfully, then repeat the sequence. Testing has shown that never more then three attempts are needed to successfully write the registers.

3.23 VOP Odd/Even field Indicator bit is incorrect

Description: The Odd/Even Indicator bit in the SAV/EAV codes within the VOP data stream is incorrect. Odd fields are indicated to be even fields and even fields are indicated to be odd fields

Implications: Because the field Indication bit is incorrect, devices connected to the VOP interface will incorrectly display the fields on a TV.

Resolution: External hardware can be used to monitor the SAV/EAV codes and invert the odd/even bit.

3.24 YUV to RGB Color Space Converter is inaccurate

Description: The YUV to RGB Color Space Converter (CSC) has rounding errors that produce errors as large as +/-5 points. See the application note *AMD Geode™ SC1200/SC1201/SC2200/SC3200 Processors YUV to RGB CSC Correction* for more details.

Implications: The YUV to RGB CSC's conversions are inaccurate. This affects the video frame buffer's quality when rendered on a CRT monitor or a TFT display. TV quality is usually unaffected because the YUV to RGB conversion is unnecessary when mixing in the YUV color space.

Resolution: The Gamma RAM in the video processor can be used to improve but not completely correct the accuracy of the conversion. See application note *AMD Geode™ SC1200/SC1201/SC2200/SC3200 Processors YUV to RGB CSC Correction Correction* for more details.

3.25 First and/or last line of a video window may flicker with flicker filter enabled

Description: Due to an architecture limitation of the flicker filter in 1/2, 1, 1/2 operation mode (F4BAR0+Memory Offset 814h[30:29] = 01), the first and last line of a video window may flicker.

The flicker filter only affects TV output. The flicker will most likely be observable when the line of video and the line of graphics are significantly different in color or intensity.

Implications: Whenever video and graphics are mixed, the first and/or last line of the video window may flicker.

Resolution: None.

4.0 Core Logic Module

4.1 GPIO status bits not always cleared

Description: Writing FFFFh to the GPST0 and GPST1 registers (F0BAR0+I/O Offset 0Ch and 1Ch, respectively) does not clear the GPIO status bits.

Implications: Interrupt routines may be called indefinitely, resulting in operating system hang-up.

Resolution: Clear these bits by reading the top-level status bits at F1BAR0+I/O Offset 02h[0].

4.2 IDE bus master buffer address must be DWORD aligned

Description: UDMA and MDMA PRD buffer address must be DWORD aligned. Several operating systems (e.g., Microsoft® Windows®, Linux) create transfer buffers that are WORD aligned. When a WORD aligned transfer buffer is used, the UDMA driver must revert back to PIO to perform the transfer.

Implications: If the address is not DWORD aligned, the bus master will function incorrectly.

Resolution: PRD address must be DWORD aligned to use UDMA. If not PIO mode must be used and performance is affected.

4.3 ACPI register access incorrectly disabled

Description: The ACPI register (accessed using F1BAR1) access is disabled after power is connected (from V_{SB} and V_{SBL}) until a falling edge occurs on PWRBTN# input.

Implications: Programming of GPWIO signals, LED signal, and ACPI registers is disabled.

Resolution: Generate a negative pulse on PWRBTN# after power is connected to the SC1200/SC1201. The PWRBTN# must revert to high state to prevent the logic from switching off after four seconds.

4.4 Simultaneously use of clock throttling and Sleep state does not work

Description: Clock throttling and Sleep state control use the same hardware and therefore can not be used simultaneously.

Implications: Clock throttling and Sleep state can not be used at the same time.

Resolution: Before entering Sleep state, turn off clock throttling (i.e., F1BAR1+I/O Offset 00h[4] = 0).

4.12 PCI burst read with wait states may hang

Description: When a master on the slow PCI bus performs a read burst that is longer than 1 KB and IRDY# is held at wait state for more than 4 clock cycles, the read burst may hang.

Implications: None. The problem does not occur with common PCI devices.

Resolution: None.

4.13 PCI special cycles not replicated from internal PCI to external PCI

Description: Special cycles such as Halt and Shutdown are not transferred to the external PCI.

Implications: This breaks PCI compliance.

Resolution: None.

4.17 Access crossing chip select boundary may affect IOCS0# and IOCS1# address mapping

Description: If a DWORD or WORD access to IOCS0# and IOCS1# crosses their upper boundary, or if the base address is not aligned to the range size, the chip selects may be active outside their address region.

Implications: None. This kind of access is not typically used.

Resolution: Do not perform an access that crosses chip select boundary, and align base address to the range size.

4.20 PCI/ISA multiplexed signals are left in an unknown state

Description: After completion of an ISA cycle, PCI/ISA multiplexed signals are left floating in an unknown state. Four PCI control signals are shared with signals in the high data byte on Sub-ISA: IRDY#/D14, TRDY#/D13, STOP#/D15, and DEVSEL#/BHE#. Only when a 16-bit I/O or 16-bit memory device is present on the Sub-ISA bus is this issue exposed. If any of the four specific data bits are being driven low at the end of the Sub-ISA cycle, and a PCI cycle immediately follows, the pull-up connected to these signals is not strong enough to pull the signal up before FRAME# is asserted. The PCI protocol is violated and unpredictable results occur including system failure. This issue does not affect 8-bit devices on the Sub-ISA bus since there are no PCI control signals on the Sub-ISA low byte, and all the signals on the Sub-ISA high byte remain high during the Sub-ISA cycle.

Implications: 16-bit I/O and 16-bit memory devices connected to the Sub-ISA bus will cause unpredictable system behavior including system failure.

Resolution: In systems that have 16-bit I/O or 16-bit memory devices, use 2K pull-ups on signals: IRDY#/D14, TRDY#/D13, STOP#/D15, and DEVSEL#/BHE#. If the four signals are pulled up within one PCI clock time, all PCI control signals will be in the correct inactive state when FRAME# asserts. Extensive testing has shown that a 2K pull-up on the four signals is more than sufficient. The 2K pull-ups do not conform to the PCI specification, however, no issues have been observed while using them.

4.22 Bandwidth limitation on PCI bus during Capture VIP

Description: If PCI master devices hold the bus for periods of 3 µs, pixels may be lost on the Capture VIP input stream (data overrun).

Implications: If PCI master devices hold the bus for periods of 3 µs, horizontal disturbances appear when viewing Capture VIP stream.

Resolution: Two possible solutions:
1. Use Direct VIP mode.

2. Design PCI bus loading so that in all possible scenarios the PCI master devices do not hold the bus for periods of 3 µs.

4.23 In DVIP mode, when the input signal is weak or not present, both video and graphics are not stable

Description: In DVIP mode, when the input signal is weak or not present (e.g., during VCR FF/RWD or channel change), since GenLock is active, the output display timings are incorrect.

Implications: Both video and graphics are not stable.

Resolution: Implement the following two mechanisms:

1. In DVIP mode, detect when the input signal goes bad and switch to VIP capture to memory, while disabling GenLock to the input signal.

2. In VIP mode, detect when the input signal recovers and switch to DVIP, while enabling GenLock to the input signal.

4.24 User Defined Traps when pointed to IDE do not work

Description: User Defined Traps do not successfully trap I/O when pointed to IDE.

Implications: User Defined Traps cannot be used to trap I/Os to IDE.

Resolution: None.

4.25 Can not boot from LPC bus using some LPC ROM

Description: The system begins the boot process from address FFFFFFF0h. The BIOS code issues a far jump instruction generally to a location below the 1 MB address range typically 000Fxxxxh. This far jump is done because the processor begins execution in “pseudo real big mode” and the far jump puts the processor in real mode where addresses above 1 MB cannot be accessed. Neither the LPC interface nor most LPC ROMs alias the addresses correctly to support the normal boot process.

Implications: Many LPC ROMs can not be used for BIOS.

Resolution: Use only LPC ROMs that alias correctly (e.g., SST 49LF020 or WinBond 49V002) or the BIOS must be on the Sub-ISA bus or behind an LPC SIO where the proper aliasing occurs.

4.26 Clearing bit 0 in the IDE Bus Master Command register does not stop the current UDMA transfer

Description: Bit 0, Bus Master Control, in the IDE Bus Master 0 and 1 Command registers (F2BAR4+I/O Offset 00h and 08h, respectively) does not function as expected. Setting the bit does enable the bus master engine to begin operation but clearing the bit while the bus master is in operation does not stop the operation.

Implications: Clearing the Bus Master Control bit can not stop the IDE bus masters.

Resolution: Use bit 2 (IDE Reset) and bit 3 (IDE Controller Reset) of the Reset Control register (F0 Index 44h) to abort the operation.

4.27 An IDE bus master transfer of exactly 64 KB cannot be done

Description: The size field in the Physical Region Descriptor (PRD) indicates that up to 64 KB can be transferred. However, setting the field with a value of zero, which is the 64 KB transfer value, does not work.

Implications: The IDE bus masters can not transfer the maximum amount expected of 64 KB using a single PRD.

Resolution: If a 64 KB transfer is expected, software must break the transfer into two 32 KB transfers by creating two PRDs.

4.29 PCI deadlock condition with IDE bus master

Description: When IDE bus master transfers are active, all I/O transactions to any IDE device are retried until the bus master has completed all its transfers. If an error occurs during an IDE bus master transfer, the IDE device prematurely terminates the transfer, however, the bus master does not terminate until reset by software. If both events happen simultaneously, a PCI deadlock condition occurs. This is because an I/O transaction is being retried and the transfer has not completed.

Implications: I/O transactions to an IDE device during a bus master transfer are at risk of creating a PCI lock condition.

Resolution: Do not invoke I/O transactions to an IDE device during bus master transfers.

4.31 PCI compliance not met on PCI bus

Description: External bus masters are sometimes required to wait longer than 16 PCI clocks before data is returned on a read.

Implications: This is a violation of the PCI bus specification protocol. This read delay also has a negative effect on PCI bus bandwidth.

Resolution: None.

4.32 IDE bus master transfers of less than 20h are not supported

Description: The IDE bus master does not properly handle data transfers of less than 20h in size.

Implications: If the operating system makes a transfer request of less than 20h bytes of data, and the IDE bus master attempts to perform the transfer, the IDE bus master will hang.

Resolution: If a request is made to transfer less than 20h bytes of data, programmed I/O should be used to perform the transfer.

4.33 PCI address stepping masters not supported on GNT1#

Description: During PCI Idle, the SC1200 will simultaneously assert GNT1# if its device is requesting the bus, and deassert another GNT# if its device is not requesting the bus or is of lower priority. This action does not meet the PCI specification.

Implications: There are two implications resulting from this issue:

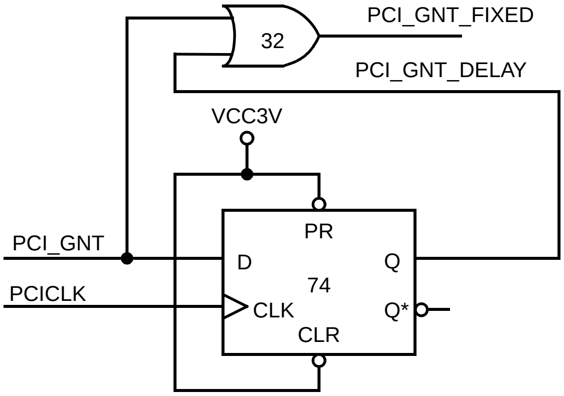
1) Address stepping PCI bus masters assert the address one clock before asserting FRAME# and with the simultaneous GNTx# asserting and GNTy# deasserting, a drive fight condition will exist on the address and command signals. If this condition continues long enough, it could cause an incorrect address or command to be signaled and unexpected system behavior would result. This applies only to addressing stepping PCI masters connected to GNT1#.

2) According to the PCI specification, a bus master that receives GNT# when the bus is Idle, can assume that the bus is owned by that bus master and can begin a cycle without verifying that the bus is actually Idle (i.e., looking for FRAME# active). Most PCI devices look at FRAME# in their PCI control state machine regardless and will not assert FRAME# if FRAME# is already asserted by another master. During bus Idle, if a bus master in the system has asserted FRAME# at the same time its GNT# is deasserted, and if a second bus master asserts FRAME# without looking for FRAME# active after its GNT# is asserted, a situation is created where both bus masters launch a PCI cycle at the same time. Both cycles fail due to bus contention and the system crashes.

Resolution:

Implication 1) Connect address stepping PCI master to GNT1# or if the PCI device has configurable address stepping, configure the device so that address stepping is not performed.

Implication 2) For the very small number of devices that have this issue: connect the device to GNT0# or apply the external hardware shown below on the GNT1#. This will prevent the issue.



GNT# Workaround

4.35 Reprogramming PIT counter mode does not reset counter latches

Description: If the PIT counter mode is reprogrammed, the counter latch flip-flop is not reset. This is incorrect behavior. This failure to reset will be exposed if software reprograms the PIT counter mode after only one read of the data port.

Implications: Generally the PIT counter mode is not reprogrammed during runtime. However, if it is reprogrammed then software generally assumes the counter latch flip-flop is also reset.

Resolution: None. Do not reprogram the PIT counter mode if only one read of the data port has occurred.

4.36 Changing Flash data width after boot does not work

Description: IBUS16, bit 14 at Offset 34h[14] of the General Configuration Block register does not work correctly.

Implications: Data bus width of Flash memory on the Sub-ISA bus can not be changed after boot.

Resolution: None.

4.38 ACPI C3 does not wake by interrupt

Description: The processor enters the C3 state, when the P_LVL3 register (F1BAR1+I/O Offset 05h) is read. It is supposed to exit this state back to the C0 state when an NMI, an unmasked interrupt, an SMI, or a bus master event (enabled via the BM_RLD bit, F1BAR1+I/O Offset 0Ch[1] = 1) occurs. The processor fails to return to C0 when and unmasked interrupt occurs.

Implications: The C3 ACPI state cannot be used if unmasked interrupts are required to return the processor back to the C0 state. In most cases unmasked interrupts are required so in general the C3 ACPI state can not be used

Resolution: Use the C1 state, Suspend on Halt (HLT) instead.

4.39 Debounce circuit does not work

Description: The power button (PWRBTN) and GPWIO[1:0] have digital debounce circuits. When enabled, the debounce circuit occasionally blocks a legitimate transition. When this happens, the desired action (generate an SMI, turn on the system) gets blocked.

Implications: With the debounce enabled, reliable system response to PWRBTN and GPWIO[1:0] transitions cannot be assured, resulting in undesirable system behavior.

Resolution: Disable debounce (F1BAR1+I/O Offset 07h[3,0]) and if required debounce the signal externally.

4.40 Left/right audio output swapped

Description: The left/right digital output data going to the AC97 codec is swapped. By convention, each audio DWORD contains 16 bits for the left speaker and 16 bits for the right speaker. The audio bus master outputs incorrectly (i.e., the 16 bits meant for the left speaker are going to the right speaker and vice-versa) according to the convention.

Implications: Applications that require sound to be appropriately positioned will not be correct. The most affected application is video playback.

Resolution: Swap the left/right speaker outputs of the AC97 codec.

4.41 IDE reset output misbehaves during software initiated reset

Description: When POR# is asserted or software enables X-Bus Warm Start (F0 Index 44h[0]), IDE_RST# is supposed to TRI-STATE. Before the TRI-STATE condition occurs the output is momentarily driven low. The IDE_RST# output does not usually reach a valid low before the TRI-STATE occurs.

Implications: The indeterminate state of IDE_RST# causes some IDE devices to fail to operate correctly after the system reset is complete. IDE drives usually have a strong enough pull-up on IDE_RST# that prevents the indeterminate state. CompactFlash drives usually do not and thus, are susceptible to the failure. CD drives often have a large capacitive load on IDE_RST# making it more difficult for a pull-up to prevent the condition.

Resolution: After POR# is de-asserted, the system boot code should assert IDE_RST# to properly reset the IDE device. When performing a warm start, software should first enable IDE Reset (F0 index 44[2]) to drive IDE_RST# to a valid low and then enable X-Bus Warm Start. This creates a reliable IDE_RST# during the system reset period

6.0 Chip I/O

6.6 Limited input voltage

Description: The voltage on all input signals should not exceed 3.6V except for 5V tolerant signals (ACK#, AFD#/DSTRB#, BUSY/WAIT#, ERR#, INIT#, PD[7:0], PE, SLCT, SLIN#/ASTRB#, STB#/WRITE#, ONCTL#, PWRCNT[2:1]).

Implications: Chip reliability cannot be guaranteed if voltage of input signals exceeds this limit.

Resolution: None.

The contents of this document are provided in connection with Advanced Micro Devices, Inc. (“AMD”) products. AMD makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice. 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.

www.amd.com

One AMD Place
P.O. Box 3453,
Sunnyvale, CA 94088-3453 USA
Tel: 408-732-2400 or 800-538-8450
TWX: 910-339-9280
TELEX: 34-6306

TECHNICAL SUPPORT
USA & Canada: 800-222-9323 or 408-749-5703
USA & Canada: PC Microprocessor: 408-749-3060

Latin America Email: spanish.support@amd.com
Argentina: 001-800-200-1111, after tone 800-859-4478
Chile: 800-532-853
Mexico: 95-800-222-9323

Europe & UK: +44–0-1276-803299
Fax: +44–0-1276-803298
France: 0800-908-621
Germany: +49–89-450-53199
Italy: 800-877224
Europe Email: euro.tech@amd.com

Far East Fax: 852-2956-0588
Japan Fax: 81-3-3346-7848