AMD Geode™ GX1 Processor Silicon Revision 8.1.1 Specification Update

1.0 Scope

This document discusses known issues of the AMD Geode™ GX1 processor, silicon revision 8.1.1. Table 1-1 provides a summary of the issues. A detailed description of each issue, its impact, and a recommended resolution/fix follow.

To determine the silicon revision of the device, printed on the chip (bottom-side of SPGA, top-side of EBGA) is the lot code number. The lot code number for silicon revision 8.1.1 is a 10-digit number with an "A" in the 5th character (e.g., V8SK**A**040AG). Note that the other characters of the lot code number may change depending upon lot number, date, etc. However, the "A" in the 5th character is constant. Software can detect this revision by reading the DIR1 Configuration register (see Configuration registers in the GX1 data book). The value read from DIR1 is 81h for silicon revision 8.1.1.

Silicon errata are tracked separately. This document pertains to silicon revision 8.1.1 only.

Note: This is revision 8.1 of this document. The change from revision 8.0 (dated May 2002) is in format only. No technical changes.

Table 1-1. Errata Summary

Issue #[1]	Description
1	Incorrect CURRENT_IP field in SMI header
2	RSM truncates page-granular CS limit
3	SDRAM CAS latency of 1 not supported
4	VIH change from 2.0V to 2.1V on FLT# input
5	PCI AD bus floats too early on some target terminated cycles, not PCI 2.1 compliant
6	Memory writes in SMI handler could have A20 in their address cleared
7	Double fault handled as general protection fault
8	Call ESP does not work
9	PCI signal SERR# asserts for two clocks, not one
10	PCI signal LOCK# ignored
11	PCI signal PERR# is floated instead of driven high on deassertion
12	PCI REQs must not go active during reset
13	CALL at beginning of Code Segment Call causes General Protection Fault
14	Self modifying code can cause PF
15	Time Stamp Counter stops during Suspend
17	WORD access to Port 23; Port 24 half of access goes off chip
19	Thermal diode does not work
20	PCI Master Latency Timer is broken
21	Data setup to PCLK does not meet specification
22	Data setup to VID_CLK does not meet specification
23	Video port limited to 133 MHz
24	Behavior of EFLAGS during INTR handling is not as expected
25	Graphics resolution of 1280x1024x16 is not supported
26	I/O write interlock
27	Time Stamp Counter rollover

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

2.0 Issues

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 is wrong (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, there is code available that allows the SMM handler to calculate the CURRENT_IP field.

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 the upper 12 bits of the original limit are lost.

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

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

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: CAS latency of 1 is not supported. The impact of this is minor, as there are very few (if any) SDRAMs that support this setting.

4. VIH change from 2.0V to 2.1V on FLT# input

Description: The Voltage Input High (VIH) on the FLT# input has been changed from 2.0V to 2.1V.

Implications: None. In most systems, FLT# is normally unused and pulled to V_{CC3}.

Resolution: None.

5. PCI AD bus floats too early on some target terminated cycles, not PCI 2.1 compliant

Description: The problem cycles occur when the processor is the target and the cycle is a read. The AD bus goes TRI-STATE when TRDY# goes inactive. The processor should TRI-STATE the AD bus when IRDY# goes inactive. However, under certain target abort conditions, IRDY# stays active longer then TRDY#, leaving the AD bus undriven for a few PCI clocks.

Implications: This breaks PCI compliance, however, there are no functional problems with this issue.

Resolution: None.

6. Memory writes in SMI handler could 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 cause the data to go 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 forcing any posted write to execute before the RSM executes.

7. Double fault handled as general protection fault

Description: A CLI is pending, causing a CPU privilege level exception. The trap gate points to a "not present" code segment. Both of these faults are contributory class exceptions and a double fault should be taken.

Implications: A double fault is not taken, however, the "not present" fault is taken. This is not a functional issue. This fault condition is a result of a coding error. A fault is taken; just not the correct fault.

Resolution: None required.

8. Call ESP does not work

Description: Call ESP instruction is broken.

Implications: This instruction exists because of the way the call register instruction is created. This instruction is never used. Using this call and managing the stack becomes extremely difficult if not impossible. Do not use this instruction.

Resolution: None.

9. PCI signal SERR# asserts for two clocks, not one

Description: SERR# is asserted for two clocks.

Implications: This breaks PCI compliance. Fault tolerant systems are the only systems affected by this issue.

Resolution: None.

10. PCI signal LOCK# 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.

11. PCI signal PERR# is floated instead of driven high on deassertion

Description: PERR# is floated instead of driven high and then set to TRI-STATE.

Implications: This breaks PCI compliance. If implemented in a system, a strong pull-up should be used on this signal.

Resolution: None.

12. 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.

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

Resolution: None.

13. CALL at beginning of Code Segment Call causes General Protection Fault

Description: A segment exists that has 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, that happens to be in the middle of the 16-byte line fetch. The limit checking hardware assumes that the limit has been crossed because the line fetch contains both the beginning and the end of the segment. The hardware is unable to discern that the actual code execution does not cross the limit, hence, causing a general protection fault to occur.

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.

14. Self modifying code can cause PF

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

Implications: Self modifying code that executes as described fails.

Resolution: None.

15. Time Stamp Counter stops during Suspend

Description: When the processor is in Suspend due to SUSP#/SUSPA# or in HALT with the "Suspend on Halt" bit set, the Time Stamp Counter stops.

Implications: This is different from other CPUs.

Resolution: None.

17. WORD access to Port 23; Port 24 half of access goes 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 (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 does go off chip.

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

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

19. Thermal diode does not work

Description: The thermal diode at pins E24 and D26 of the EBGA package and F36 and E37 of the SPGA package do not work. Treat these signals as no connects.

Implications: The thermal diode cannot be used.

Resolution: None. Do not use this feature.

20. PCI Master Latency Timer is broken

Description: PCI register 0Dh, the Master Latency Timer, is broken. Setting this register to any value other then 00h results in a system hang.

Implications: This breaks PCI compliance, however, the Master Latency Timer is typically not used.

Resolution: None. Do not use this feature.

21. Data setup to PCLK does not meet specification

Description: Symbol t4 in Table 6-19 (Video Interface Signals) in the GX1 data book is out of specification. The timing of t4 is 5 ns, not 3.8 ns.

Implications: PCLK is limited to 135 MHz instead of 157 MHz. This limits maximum resolution to 1280x1024x75 Hz instead of 1280x1024x85 Hz.

Resolution: None.

22. Data setup to VID_CLK does not meet specification

Description: Symbol t8 in Table 6-19 (Video Interface Signals) in the GX1 data book is out of specification. The timing of t8 is 5 ns, not 3.8 ns.

Implications: VID_CLK is limited to 133 MHz (266/2) instead of 150 MHz (300/2). VID_CLK is created by dividing the core frequency by 2 or 4. With this limitation and a core frequency of 300 MHz, VID_CLK can only be divided by 4. With VID_CLK limited to 75 MHz, the video window cannot be used if the graphics resolution is higher than 1024x768x85 Hz.

Resolution: None.

23 Video port limited to 133 MHz

Description: There is currently no I/O companion solution that supports 150 MHz on the video port.

Implications: VID_CLK is created by dividing the core frequency by 2 or 4. With this limitation and a core frequency of 300 MHz, VID_CLK can only be divided by 4. With VID_CLK limited to 75 MHz, the video window cannot be used if the graphics resolution is 1280x1024x85 Hz. 1280x1024x75 Hz and lower function correctly.

Resolution: None.

24. Behavior of EFLAGS during INTR handling is not as expected.

Description: If an IRQ occurs during EFLAGS style CPU ID support detection, bit 22 gets cleared. This is different from an Intel CPU.

Implications: Possible compatibility problems.

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

25. Graphics resolution of 1280x1024x16 is not supported

Description: If the horizontal resolution is greater than 1024, the bits per pixel is limited to 8 instead of 16.

Implications: The standard resolution of 1280x1024x16 is not supported.

Resolution: None.

26 I/O write interlock

Description: I/O writes are non-postable instructions. Code execution is not supposed to continue until the I/O write is completed. An I/O write instruction that is executed on the PCI bus can be retried in a process called delayed transaction. This allows external PCI bus masters to gain control of the bus, even though a long cycle is in progress to a slow device, such as ISA bus devices. If a bus master gains control of the PCI bus during a delayed I/O write and transfers data, the CPU core is signaled that the delayed I/O write has completed. This allows the CPU core to continue inadvertently.

Implications: The vast majority of I/O writes are not affected by this issue, because they do not care if code continues to execute even though they have not completed. However, in extremely rare instances, unexpected system behavior can result.

Resolution: If a critical I/O write exists, execute a non-critical I/O read immediately after the I/O write.

27 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.

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
USA & Canada Email: pcs.support@amd.com

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