



## VoyagerGX Rev. AB Errata Information

### Introduction

This document describes the errata information for the VoyagerGX (SM501) Rev. AB silicon. All the issues outlined in Errata for the earlier chip revisions have been fixed in Rev. AB silicon.

### Errata Description Summary

#	DESCRIPTION
AA-1.	UART Cannot Run Higher Than 38.4K Baud Rate
AB-2.	PCI slave mode data burst does not work on memory mapped register space
AB-3.	PCI Master Mode Only Runs at 72MHz for 33Mhz PCI
AB-4.	Hardware Cursor Causes Garbage
AB-5.	Restrictions on selection of engine clock and memory clock
AB-6.	Command Interpreter's stop/finish command do not work properly
AB-7.	Wake Up from sleep by detecting "CS" pin does not Work reliably
AB-8.	12-bit CSTN Display Interface
AB-9.	Restriction on CPU bus clock/ SM501's host clock ratio (for SM501 to CPU bus directly interface)
AB-10.	USB Keyboard / Mouse Hang on CPU Local Bus when using the SM501' internal memory as the USB buffer
AB-11.	Cannot Program Hardware I <sup>2</sup> C
AB-12.	NAND Tree Scan Test Not Available
AB-13.	PCI slave mode data burst does not work on memory transfer

### Fixed Issues Summary

#	DESCRIPTION
AA-11	8-bit Digital CRT and 8-bit ZV Interface Can't be Used Simultaneously

### Revision History

Version	Date	Note
0.1	1/9/2004	First Draft for Rev. AB silicon.
0.2	1/16/2004	Add AB-11
0.3	1/28/2004	Modified AA-1 with workaround.
0.4	7/14/2004	Add AB-12
0.5	10/12/2004	Add AB-13

This document is for information only and is subject to change without notice. Silicon Motion®, Inc. reserves the right to make changes in the product design without reservation and without notice to its users.



Notes:

1. The "host clock" in the following is referring to the "system DRAM clock" in SM501 data book, which is controlled by registers - base+0x000068(and 6C), bits[20:16]
2. The "engine clock" in the following is referring to the "Mclk" in SM501 data book, controlled by register base + 0x000044(and 4C), bits[12:8]
3. The "memory clock" in the following is referring to the "M1xclk" in SM501 data book, controlled by register base + 0x000044(and 4C), bits[4:0]

### AA-1. UART Cannot Run Higher Than 38.4K Baud Rate

<b>Description:</b>	UART does not function properly if the baud rate is faster than 38400 bps.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	None
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	The baud rate speed is limited by the UART block internal source clock frequency.
<b>Workaround:</b>	UART can support up to 115200 bps baud rate for standard serial port or 256000 bps for designed serial port. In order to reach these high baud rates, the SM501's 24 MHz crystal input must be changed to 22.1184 MHz or 24.576 MHz, respectively. However, this crystal input frequency does not meet SM501 USB Host port requirement as 24 MHz and will cause the USB Host port malfunction.  Will be fixed in the future product family.

### AB-2. PCI slave mode data burst does not work on memory mapped register space

<b>Description:</b>	Memory mapped register can not accept PCI burst mode. This will also affect normal memory burst since the hardware does not recognize a PCI cycle is a register access or memory access.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	For future performance tuning driver will control the burst on/off for larger memory source copy operation.
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	Hardware does not recognize a PCI cycle is a register access or memory access.
<b>Workaround:</b>	Normally turn the PCI burst off. Will control the PCI burst on/off in the driver for larger memory data transfer in future performance tuning.



### **AB-3. PCI Master Mode Only Runs at 72MHz for 33Mhz PCI**

---

<b>Description:</b>	For a 33Mhz PCI system, the engine clock setting is limited to max 72Mhz for proper PCI master cycle operation. For a 66Mhz PCI system, the engine clock setting can be set up to 125Mhz.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	None
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	Caused by two clock domain synchronization issue.
<b>Workaround:</b>	Slow down the engine clock to 72Mhz for a 33Mhz PCI system if PCI master mode has to be used. Will be fixed in the future product family.

### **AB-4. Hardware Cursor Causes Garbage**

---

<b>Description:</b>	The hardware cursor will cause display garbage when moving vertically.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	Device driver need to add patch.
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	The Y position did not synchronize with VSYNC.
<b>Workaround:</b>	<p>Software Workaround: allow the hardware cursor Y position to change only during VSYNC.</p> <p>The software patch works perfectly without causing performance degradation.</p> <p>Will be fixed in the future product family.</p>



---

## AB-5. Restrictions on selection of engine clock and memory clock

---

<b>Description:</b>	The engine clock and memory clock need to be derived from a same PLL source. Otherwise memory controller may hang up causing vertical bar appears on screen.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	The clock setting for the engine clock and memory clock need to be selected from a same PLL source.
<b>Affected Registers:</b>	Base+0x00004C & Base+0x000044, Power mode0&1 clock registers
<b>Root Cause:</b>	The problem is caused by crossing clock domain between the engine clock and memory clock. If the two clocks are derived from a same PLL, the clock can be set at different divider ratio but maintain a fixed relationship, then the problem can be avoided.
<b>Workaround:</b>	Software workaround: software selects a same PLL source for both the engine clock and memory clock.

---

## AB-6. Command Interpreter's stop/finish command do not work properly

---

<b>Description:</b>	Use of the command interpreter's stop/restart command during data transfer will cause data corruption.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	Can be work around in software driver.
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	Caused by HW implementation of the "stop" command control logic.
<b>Workaround:</b>	Software work around by avoid using "stop/restart" combination in data transfer is proven to be reliable in WinCE driver. Please refer to SM501 DDK for different approach on CMD interpreter programming.  Will be fixed in the future product family.



## **AB-7. Wake Up from sleep by detecting “CS” pin does not Work reliably**

<b>Description:</b>	When SM501 detect “CS” pin to wake up from sleep mode, the chip may fail to wake up to cause system to hang.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	Need software patch.
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	When wake up from sleep mode (all SM501 PLL are shut down), the host clock is not gated (always enable). When PLL is turned on upon wake up, the clock derived from the PLL may not be stable, which cause the internal state machine to go to wrong state.
<b>Workaround:</b>	Software Workaround: During wake up from sleep mode, software detects CS command from CPU and adds two level delays until the internal PLL stabilize; then issues the CS command to SM501.  Will be fixed in the future product family.

## **AB-8. 12-bit CSTN Display Interface**

<b>Description:</b>	12-bit CSTN panel interface does not function properly.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	None
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	CSTN clock is not functioning properly. When 12-bit CSTN is selected, only 8-bit CSTN clock is sent out instead of 12-bit CSTN clock.
<b>Workaround:</b>	External logic to convert 8-bit clock to 12-bit CSTN clock.  Will be fixed in the future product family.



## **AB-9. Restriction on CPU bus clock/ SM501's host clock ratio (for SM501 to CPU bus directly interface)**

<b>Description:</b>	SM501 host interface can run asynchronously respect CPU bus clock. The SM501's host clock need to be at least 95% or greater of the CPU bus clock. For example, if the CPU bus clock is 100Mhz, the SM501 host clock need to be $\geq 96\text{Mhz}$ .
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	Always program SM501 host clock frequency before access SM501 memory
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	Host address did not have enough hold time if the host clock is programmed much slower than the CPU bus clock. This is only apply to the SM501 memory access cycles.
<b>Workaround:</b>	Software Workaround: always program the base+0x000068 register first to set host clock within 95% or greater speed vs. the CPU bus clock.  Will be fixed in the future product family.

## **AB-10. USB Keyboard / Mouse Hang on CPU Local Bus when using the SM501' internal memory as the USB buffer**

<b>Description:</b>	When device driver select SM501's internal memory as USB buffer, the USB keyboard or mouse hang up randomly while the system is still running. This only happens when SM501 interface with CPU local bus directly.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	None
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	When CPU reads SM501 through CPU local bus, it will take the data from SM501's cache-buffer if the address is a hit, or take the data directly from the local memory if the address is a miss. The USB controller of SM501 constantly accesses the USB portion of local memory through internal DMA with current data and status. The cache-buffer on the local bus does not have the ability to snoop the internal DMA activities. Therefore, the data read from the cache-buffer in hit may not be the current updated one from the local memory. This causes the USB keyboard / mouse hangs.
<b>Workaround:</b>	Software Workaround: Do not use SM501 internal memory as USB buffer, instead, use system as the USB buffer. If using SM501's internal memory as USB buffer, a periodic reading in a spare address of local memory in 1ms is added to the OS USB driver, from which the CPU reading will always get the current updated data from the local memory. The dummy read may cause minor bandwidth drop on the local bus but no performance hit has been observed under WinCE.



## **AB-11. Cannot Program Hardware I<sup>2</sup>C**

---

<b>Description:</b>	The Hardware I <sup>2</sup> C cannot be programmed.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	None
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	SM501 always returns 1 for status bits on read.
<b>Workaround:</b>	Use software I <sup>2</sup> C. Will be fixed in the future product family.

## **AB-12. NAND Tree Scan Test Not Available**

---

<b>Description:</b>	The NAND Tree Scan Test cannot be preformed.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	None
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	The NAND Tree Scan Test is broken internally; therefore, there is no NAND Tree Scan Test Description available.
<b>Workaround:</b>	NAND Tree Scan Test is only for checking the device soldering quality to the PCB; furthermore, NAND Tree Scan Test will not affect any other SM501 function. Will be fixed in the future product family.

## **AB-13. PCI slave mode data burst does not work on memory transfer**

---

<b>Description:</b>	PCI burst mode memory transfer did not work properly.
<b>Hardware Consideration:</b>	None
<b>Software Consideration:</b>	Software should not enable PCI burst mode.
<b>Affected Registers:</b>	None
<b>Root Cause:</b>	When the PCI data FIFO was almost full, the PCI logic did not stop the burst cycle by issuing STOP signal to the PCI master fast enough so some data is lost.
<b>Workaround:</b>	None.



## Issues Fixed in Rev. AB

### AA-11. 8-bit Digital CRT and 8-bit ZV Interface Can't be Used Simultaneously

**Description:** When GPIO[63:56] are set to drive TV display, enabling the 8-bit ZV Port will cause TV display blank. However, enabling the 16-bit ZV Port will not cause TV display blank but the ZV capture image is incorrect or broken.

**Affected Registers** 0x090000 [9]

**Root Cause:** The definition of the 8/16-bit ZV Port interfaces between the Capture Control block and the internal MUX are reversed.

<u>Register 0x90000[9]</u>	<u>Capture Control Block</u>	<u>Internal MUX</u>
0:	16-bit	8-bit
1:	8-bit	16-bit

When 0x090000[9] is set to 8-bit ZV Port interface, the MUX considers it as 16-bit interface and enables GPIO[63:56] as input mode. Therefore, if GPIO[63:56] are driving TV display, enabling register 0x090000[9] for 8-bit ZV Port interface simultaneously will cause the TV display blank.