Stellaris® LM3S5732 RevA0 Errata

This document contains known errata at the time of publication for the Stellaris LM3S5732 microcontroller. The table below summarizes the errata and lists the affected revisions. See the data sheet for more details.

See also the ARM® Cortex™-M3 errata, ARM publication number PR326-PRDC-009450 v2.0.

Table 1. Revision History

Date	Revision	Description
March 2012	2.9	■ Added issue “The USB PLL may fail to lock after reset” on page 20.
September 2011	2.8	■ Added issue “Boundary scan is not functional” on page 5.
August 2011	2.7	■ Clarified issue “Writes to certain Hibernation module registers sometimes fail” on page 8. ■ Added issue “Writes to Hibernation module registers may change the value of the RTC” on page 9. ■ Clarified issue “General-purpose timer 16-bit Edge Count or Edge Time mode does not load reload value” on page 14 to include Edge-Time mode. ■ Added issue “Retriggering a sample sequencer before it has completed the current sequence results in continuous sampling” on page 16. ■ Added issue “ADC inputs have the incorrect channel assignments” on page 17. ■ Added issue “USB Host controller may not be used to communicate with a low-speed Device when connected through a hub” on page 19. ■ Minor edits and clarifications.
September 2010	2.6	■ Added issue “MOSC valid detect circuit should not be enabled” on page 6. ■ Removed the "ROM_I2CMasterErr function is incorrect" issue because the data sheet has been changed such that the ERROR bit no longer is set when the ARBLST bit is set. ■ Minor edits and clarifications.
July 2010	2.5	■ Added information that “No interrupt generated with software-triggered DMA transfer using channel 30” on page 10 is fixed for parts with certain date codes. ■ Added issue “PB0 and PB1 have permanent internal pull-down resistance” on page 12. ■ Added issue “The RTRIS bit in the UARTRIS register is only set when the interrupt is enabled” on page 17. ■ Clarified issue “Violating USB VBUS in-rush current specifications may cause USB controller to hang” on page 18. ■ Minor edits and clarifications.
June 2010	2.4	■ Added issue “External reset does not reset the XTAL to PLL Translation (PLLCFG) register” on page 6.
May 2010	2.3	■ Removed issue "Hibernation Module 4.194304-MHz oscillator supports a limited range of crystal load capacitance values" as it does not apply to this part. ■ Minor edits and clarifications.

Date	Revision	Description
April 2010	2.2	■ Added issue "Hibernation Module 4.194304-MHz oscillator supports a limited range of crystal load capacitance values." ■ Minor edits and clarifications.
April 2010	2.1	■ Removed issue "Setting Bit 7 in I2C Master Timer Period (I2CMTPR) register may have unexpected results". The data sheet description has changed such that this is no longer necessary. ■ Minor edits and clarifications.
February 2010	2.0	■ Added issue “The General-Purpose Timer match register does not function correctly in 32-bit mode” on page 15. ■ Added issue "Setting Bit 7 in I2C Master Timer Period (I2CMTPR) register may have unexpected results".
Jan 2010	1.9	■ "Hard Fault possible when waking from Sleep or Deep-Sleep modes and Cortex-M3 Debug Access Port (DAP) is enabled" has been removed and the content added to the LM3S5732 data sheet. ■ Added issue “The µDMA controller fails to generate capture mode DMA requests from Timer A in the Timer modules” on page 11. ■ Added issue “A spurious DMA request is generated when the timer rolls over in Input-Edge Time mode” on page 15.
Dec 2009	1.8	Started tracking revision history.

Table 2. List of Errata

Erratum Number	Erratum Title	Module Affected	Revision(s) Affected
1.1	Subsequent attempts to mass erase the Flash memory following a locked device recovery will fail	JTAG and Serial Wire Debug	A0
1.2	JTAG INTEST instruction does not work	JTAG and Serial Wire Debug	A0
1.3	Boundary scan is not functional	JTAG and Serial Wire Debug	A0
2.1	Debugger cannot halt processor when in Sleep mode	System Control	A0
2.2	I/O buffer 5-V tolerance issue	System Control	A0
2.3	External reset does not reset the XTAL to PLL Translation (PLLCFG) register	System Control	A0
2.4	MOSC valid detect circuit should not be enabled	System Control	A0
3.1	Hibernation module may have higher current draw than specified in data sheet under certain conditions	Hibernation Module	A0
3.2	Hibernation module HIB pin is not an open-drain output	Hibernation Module	A0
3.3	Hibernation module registers are cleared by POR if not in Hibernate mode	Hibernation Module	A0
3.4	Hibernation Raw Interrupt Status (HIBRIS) register may not properly indicate Hibernation wake cause	Hibernation Module	A0
3.5	In Hibernate mode, the microcontroller may be parasitically powered by external USB signals	Hibernation Module	A0

Erratum Number	Erratum Title	Module Affected	Revision(s) Affected
3.6	Writes to certain Hibernation module registers sometimes fail	Hibernation Module	A0
3.7	Writes to Hibernation module registers may change the value of the RTC	Hibernation Module	A0
4.1	FMPPE and FMPRE registers cannot be committed to non-volatile storage	Flash Controller	A0
5.1	ROM_SSIConfigSetExpClk function is incorrect	ROM	A0
6.1	No interrupt generated with software-triggered DMA transfer using channel 30	µDMA	A0
6.2	The µDMA controller fails to generate capture mode DMA requests from Timer A in the Timer modules	µDMA	A0
7.1	GPIO commit register control not working as expected	GPIO	A0
7.2	PB0 and PB1 have permanent internal pull-down resistance	GPIO	A0
8.1	General-purpose timer 16-bit Edge Count or Edge Time mode does not load reload value	General-Purpose Timers	A0
8.2	The General-Purpose Timer match register does not function correctly in 32-bit mode	General-Purpose Timers	A0
8.3	A spurious DMA request is generated when the timer rolls over in Input-Edge Time mode	General-Purpose Timers	A0
9.1	Use of "Always" triggering for ADC Sample Sequencer 3 does not work	ADC	A0
9.2	Incorrect behavior with timer ADC triggering when another timer is used in 32-bit mode	ADC	A0
9.3	ADC hardware averaging produces erroneous results in differential mode	ADC	A0
9.4	Retriggering a sample sequencer before it has completed the current sequence results in continuous sampling	ADC	A0
9.5	ADC inputs have the incorrect channel assignments	ADC	A0
10.1	The RTRIS bit in the UARTRIS register is only set when the interrupt is enabled	UART	A0
11.1	I^2C Slave Masked Interrupt Status (I2CSMIS) register bits are incorrect	I2C	A0
12.1	Device Capabilities 6 (DC6) register incorrectly specifies USB OTG functionality	USB	A0
12.2	Violating USB VBUS in-rush current specifications may cause USB controller to hang	USB	A0
12.3	PB0 and PB1 pins may not be used for GPIO or peripheral functions if USB Host or Device mode functionality is used	USB	A0
12.4	Transfer may stall when size of µDMA transfer does not match USB FIFO	USB	A0
12.5	USB Host controller may not be used to communicate with a low-speed Device when connected through a hub	USB	A0

Erratum Number	Erratum Title	Module Affected	Revision(s) Affected
12.6	The USB PLL may fail to lock after reset	USB	A0
13.1	Certain pins do not fully comply with the JEDEC ESD standard	Electrical Characteristics	A0

1 JTAG and Serial Wire Debug

1.1 Subsequent attempts to mass erase the Flash memory following a locked device recovery will fail

Description:

If a user performs a locked device recovery as described in the data sheet, subsequent attempts to mass erase the Flash memory will fail. The failure status in the **Flash Controller Raw Interrupt Status (FCRIS)** register following the attempted mass erase operation has the `Access Raw Interrupt Status (ARIS)` bit set, indicating an access error. The access error is generated because the recovery sequence incorrectly restores **Flash Memory Protection Enable 2 (FMPPE2)** and **FMPPE3** to the value 0xFFFF.FFFF. During subsequent mass erase attempts, the logic (that ensures write-protected Flash memory is not erased) incorrectly evaluates the protected condition and inhibits the erase operation.

Note: This erratum may affect the ROM-resident boot loader because this module utilizes Flash memory mass erase. The boot loader will not function following a recovery sequence because the boot loader attempts a mass erase, detects the error code, and terminates.

Workaround:

There is a dynamic workaround in which the **FMPPEn** registers are written to properly set the **FMPPEn** state before a mass erase is attempted. The **FMPPEn** registers must be written to the values indicated in the following table based on the size of Flash memory available on the microcontroller. The **FMPPEn** register must be written before each mass erase operation that is attempted.

Table 3. FMPPEn Default Values Based on Flash Memory Size

Flash Size (KB)	FMPPE3	FMPPE2	FMPPE1	FMPPE0
128	0x0000.0000	0x0000.0000	0xffff.ffff	0xffff.ffff
96	0x0000.0000	0x0000.0000	0x0000.ffff	0xffff.ffff
64	0x0000.0000	0x0000.0000	0x0000.0000	0xffff.ffff
32	0x0000.0000	0x0000.0000	0x0000.0000	0x0000.ffff

Silicon Revision Affected:

A0

1.2 JTAG INTEST instruction does not work

Description:

The JTAG INTEST (Boundary Scan) instruction does not properly capture data.

Workaround:

None.

Silicon Revision Affected:

A0

1.3 Boundary scan is not functional

Description:

The boundary scan is not functional on this device.

Workaround:

None.

Silicon Revision Affected:

A0

Fixed:

Fixed on devices with date codes of 0x1A (October, 2011) or later.

2 System Control

2.1 Debugger cannot halt processor when in Sleep mode

Description:

When the processor is in Sleep mode, the debugger is unable to bring the processor out of Sleep mode in order to initiate a debug session.

Workaround:

Do not use Sleep mode when attempting to debug the application.

Silicon Revision Affected:

A0

2.2 I/O buffer 5-V tolerance issue

Description:

GPIO buffers are not 5-V tolerant when used in open-drain mode. Pulling up the open-drain pin above 4 V results in high current draw.

Workaround:

When configuring a pin as open drain, limit any pull-up resistor connections to the 3.3-V power rail.

Silicon Revision Affected:

A0

2.3 External reset does not reset the XTAL to PLL Translation (PLLCFG) register

Description:

Performing an external reset (anything but power-on reset) reconfigures the XTAL field in the **Run-Mode Clock Configuration (RCC)** register to the 6 MHz setting, but does not reset the **XTAL to PLL Translation (PLLCFG)** register to the 6 MHz setting.

Consider the following sequence:

1. Performing a power-on reset results in XTAL = 6 MHz and **PLLCFG** = 6 MHz
2. Write an 8 MHz value to the XTAL field results in XTAL = 8 MHz and **PLLCFG** = 8 MHz
3. $\overline{\text{RST}}$ asserted results in XTAL = 6 MHz and **PLLCFG** = 8 MHz

In the last step, **PLLCFG** was not reset to its 6MHz setting. If this step is followed by enabling the PLL to run from an attached 6-MHz crystal, the PLL then operates at 300MHz instead of 400MHz. Subsequently configuring the XTAL field with the 8 MHz setting does not change the setting of **PLLCFG**.

Workaround:

Set XTAL in **PLLCFG** to an incorrect value, and then to the desired value. The second change updates the register correctly. Do not enable the PLL until after the second change.

Silicon Revision Affected:

A0

2.4 MOSC valid detect circuit should not be enabled

Description:

If bit 0 of the **Main Oscillator Control (MOSCCTL)** register is set, the microcontroller is immediately reset and control is transferred to the NMI handler, even if the MOSC is functioning correctly.

Workaround:

None.

Silicon Revision Affected:

A0

3 Hibernation Module

3.1 Hibernation module may have higher current draw than specified in data sheet under certain conditions

Description:

If a battery voltage is applied to the VBAT power pin prior to power being applied to the VDD power pins of the device, the current draw from the VBAT pin is greater than expected. The current may be as high as 1.6 mA instead of the data sheet specified 17 µA. The condition exists until power is applied to the VDD pin. Once the VDD pin has been powered, the VBAT current draw functions as

expected. The VDD pin can then be powered up and down as required and the VBAT pin current specification is maintained.

Workaround:

The VBAT pin higher-than-specified current draw condition can be avoided if the microcontroller's VDD power pins are powered on prior to the time a battery voltage is initially applied to the VBAT pin.

Silicon Revision Affected:

A0

3.2 Hibernation module HIB pin is not an open-drain output

Description:

Because the HIB pin is not an open-drain output, the pin is capable of sourcing current to the power supply circuit that it controls. Under certain conditions, this configuration may result in the Hibernation module drawing more current than expected.

If the voltage on the circuit connected to HIB drops below V_{BAT} before HIB is asserted, current may be sourced out of the HIB pin, increasing Hibernation module current to approximately 300 µA instead of 16 µA. This condition also occurs if the Hibernation module attempts to wake but the power supply circuit remains below V_{BAT}.

Workaround:

Add a diode between the HIB pin and the power supply to prevent current being sourced by the Hibernation module. The cathode of the diode should be connected to HIB signal. A diode with a low reverse current, such as Panasonic MA2J728, is recommended.

Silicon Revision Affected:

A0

3.3 Hibernation module registers are cleared by POR if not in Hibernate mode

Description:

The Hibernation module registers are reset by a POR event if the module is not in Hibernate mode. This condition disables the module and stops the 32-kHz clock.

Workaround:

To preserve Hibernation module settings, always enter Hibernate mode before removing power. Note that the battery-backed memory is not cleared and is valid as long as V_{BAT} voltage remains within specification.

Silicon Revision Affected:

A0

3.4 Hibernation Raw Interrupt Status (HIBRIS) register may not properly indicate Hibernation wake cause

Description:

The **Hibernation Raw Interrupt Status (HIBRIS)** register may not properly indicate the event that caused the wake-up from hibernation. After the first wake-up event, the external wake (EXTW) status bit will get set incorrectly each time a wake-up event is triggered, regardless of the wake-up source.

Workaround:

If the EXTW bit is the only bit set, the cause is an external wake-up. If the EXTW bit and another bit is set, the wake-up source is indicated by the other bit.

Silicon Revision Affected:

A0

3.5 In Hibernate mode, the microcontroller may be parasitically powered by external USB signals

Description:

If a USB cable is still connected when the device is in Hibernation mode, the part may be parasitically powered by external voltages on USB signals. These signals include the USB bidirectional differential data pins USB0DP and USB0DM, and possibly the USB VBUS supply if connected to PB1, and the pull-up on PB0 if the part is used as a USB device.

Workaround:

If Hibernate mode is required, it may be necessary to insert a switching circuit in the USB signals to provide isolation during power down. The switching circuit should be controlled by the HIB pin.

Silicon Revision Affected:

A0

3.6 Writes to certain Hibernation module registers sometimes fail

Description:

Due to a synchronization issue with the independent clock domain of the Hibernation module, writes to certain registers may sometimes fail, even though the WRC bit in the **HIBCTL** register is set after the write occurs. Registers affected include **HIBRTCC**, **HIBRTCM0**, **HIBRTCM1**, **HIBRTCLD**, **HIBRTCT**, and **HIBDATA**.

Workaround:

After performing a write to any Hibernation module register or battery-backed memory, read the contents back and verify that they are correct. If they are incorrect, perform the write operation again.

Silicon Revision Affected:

A0

3.7 Writes to Hibernation module registers may change the value of the RTC

Description:

If the Hibernation module's RTC counter is active, any write to certain Hibernation module registers that occurs while the RTC counter is changing from the current value to the next can cause corruption of the RTC counter stored in the **HIBRTCC** register. Registers affected are: **HIBRTCC**, **HIBRTCM0**, **HIBRTCM1**, **HIBRTCLD**, **HIBRTCT**, and **HIBDATA**.

Workaround:

The user application must guarantee that writes to the affected Hibernation module registers cannot occur on the RTC counter boundary. Any initial configuration of the affected Hibernation module registers must be done before enabling the RTC counter.

There are two ways to update affected Hibernation Module registers after initial configuration:

1. Use the Hibernation RTC match interrupt to perform writes to the affected Hibernation module registers. Assuming the interrupt is guaranteed to be serviced within 1 second, this technique provides a mechanism for the application to know that the RTC update event has occurred and that it is safe to write data to the affected Hibernation module registers. This method is useful for applications that don't require many writes to Hibernation module registers.

2. Set up a secondary time-keeping resource to indicate when it is safe to perform writes to the affected Hibernation module registers. For example, use a general purpose timer in combination with the Hibernation RTC match interrupt. In this scenario, the RTC match interrupt is used to both update the match register value and enable the general purpose timer in one-shot mode. The timer must be configured to have a maximum time-out period of less than 1 second. In this configuration, a global variable is used to indicate that it is safe to perform writes to the affected Hibernation module registers. When the one-shot timer times out, the timer interrupt updates the global variable to indicate that writes are no longer safe. This procedure is repeated on every RTC match interrupt.

Silicon Revision Affected:

A0

4 Flash Controller

4.1 FMPPE and FMPRE registers cannot be committed to non-volatile storage

Description:

The Flash memory protection provided by the **Flash Memory Protection Read Enable n (FMPREn)** and **Flash Memory Protection Program Enable n (FMPPEn)** registers does not function correctly. Do not commit any of these registers or mass erase will not function.

This does not affect the Flash memory protection provided by the disabling of the JTAG/SWD port.

Workaround:

None.

Silicon Revision Affected:

A0

5 ROM

5.1 ROM_SSIConfigSetExpClk function is incorrect

Description:

If a non-Motorola format was specified in a call to the ROM_SSIConfigSetExpClk function, two lower bits of a clock divisor register could be corrupted. This corruption results in a small error in the actual clock rate.

Workaround:

Use the StellarisWare SSIConfigSetExpClk function in Flash memory.

Silicon Revision Affected:

A0

6 µDMA

6.1 No interrupt generated with software-triggered DMA transfer using channel 30

Description:

When performing a software-triggered data transfer using µDMA channel 30, no interrupt is triggered when the transfer completes.

Workaround:

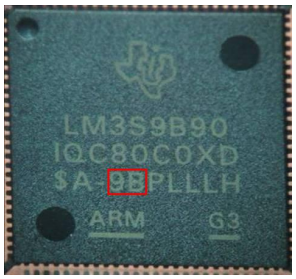
Use any of the other channels (except for channel 30 and 31) that are not already assigned to a peripheral. See the "DMA Channel Assignments" table in the *Micro Direct Memory Access (µDMA)* chapter in the data sheet.

Silicon Revision Affected:

A0

Fixed:

This issue is fixed on parts with a date code of 06 (June 2010) or later. To determine the date code of your part, look at the third line in the part markings, at the fourth and fifth characters following the dash (highlighted in red below). The first number after the dash indicates the last decimal digit of the year. The next character indicates the month, in hexadecimal. So, in the below example, the 9B indicates a date code of November 2009.



The table below shows some example date codes:

Date Code	Date
9B	November 2009
9C	December 2009
01	January 2010
02	February 2010
03	March 2010
04	April 2010
05	May 2010
06	June 2010
07	July 2010
08	August 2010
09	September 2010
0A	October 2010
0B	November 2010
0C	December 2010

6.2 The µDMA controller fails to generate capture mode DMA requests from Timer A in the Timer modules

Description:

The µDMA controller fails to generate DMA requests from Timer A in the General-Purpose Timer modules when in the Event Count and Event Time modes.

Workaround:

Use Timer B.

Silicon Revision Affected:

A0

7 GPIO

7.1 GPIO commit register control not working as expected

Description:

The commit control registers provide a layer of protection against accidental programming of critical hardware peripherals, especially the JTAG/SWD pin functionality. Writes to protected bits of the **GPIO Alternate Function Select (GPIOAFSEL)**, **GPIO Digital Enable (GPIODEN)**, and **GPIO**

Pull-up Select (GPIOPUR) registers are not committed to storage unless the **GPIO Lock (GPIOLOCK)** register has been unlocked and the appropriate bits of the **GPIO Interrupt Clear (GPIOCR)** register have been set. Registers **GPIODEN** and **GPIOPUR** should be protected by this mechanism but are not.

Workaround:

Extra caution should be taken by software when modifying the **GPIODEN** and **GPIOPUR** registers as they are not protected by the **GPIOCR** commit register mechanism. Writes to the register bits that affect the JTAG/SWD pins (PC0-PC3) should be done with great care as this interface may become permanently disabled if done incorrectly. The NMI pin (PB7) is not protected either.

Silicon Revision Affected:

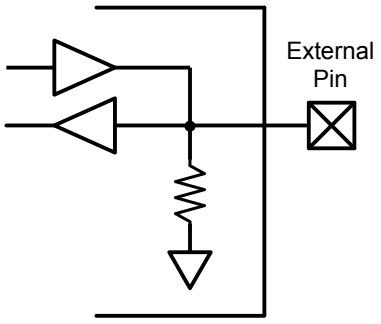
A0

7.2 PB0 and PB1 have permanent internal pull-down resistance

Description:

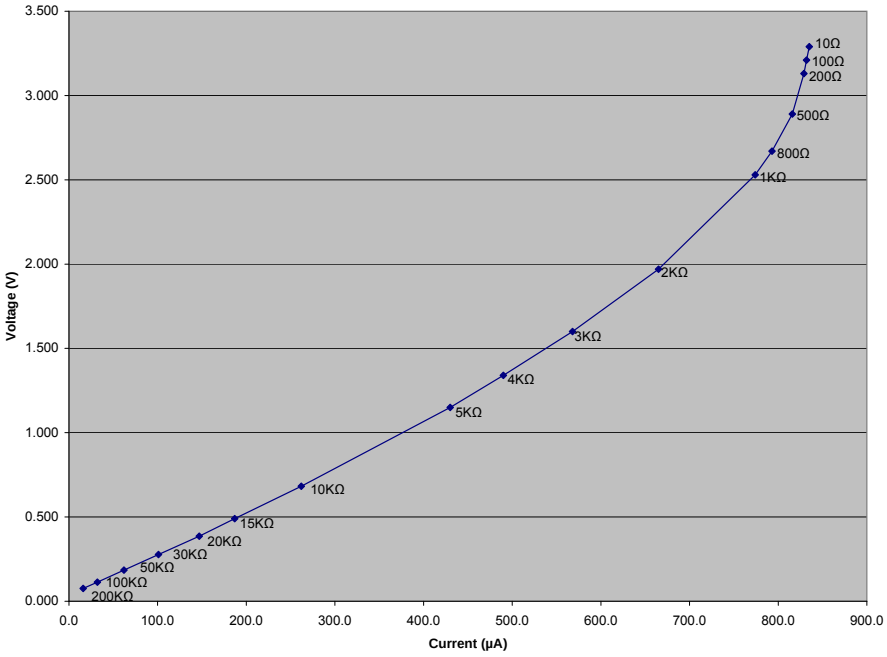
Regardless of their configuration, PB0 and PB1 have an internal pull-down resistance. The internal structure of these pins is shown in Figure 1 on page 12.

Figure 1. Internal Structure of PB0 and PB1



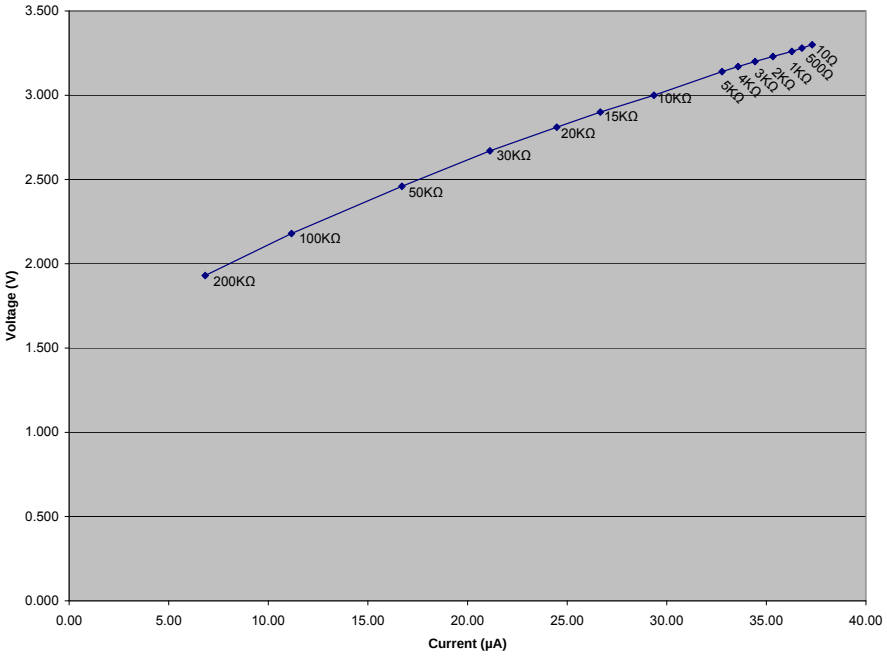
The characteristic of the pull-down for PB0 is shown in Figure 2 on page 13. The data labels for each data point show the external pull-up resistance in Ohms.

Figure 2. Voltage vs. Current for Various External Pull-up Resistors on PB0



The characteristic of the pull-down for PB1 is shown in Figure 3 on page 14. The data labels for each data point show the equivalent resistance in Ohms.

Figure 3. Voltage vs. Current for Various External Pull-up Resistors on PB1



Workaround:

When either of these pins is configured as an input, the external circuit must provide enough drive strength to over-drive the internal pull-down and achieve the necessary V_{IH} voltage level. If an external pull-up resistor is used, it must have a low value such as ~1 KΩ or less for PB0 and ~50 KΩ or less for PB1.

When either of these pins is configured as an output, the drive current needed to over-drive the internal pull-down resistance must be subtracted from the drive capabilities of the pin. In some applications, it may be necessary to select a higher drive strength (such as 4 mA instead of 2 mA) to achieve an acceptable output voltage on PB0.

Silicon Revision Affected:

A0

8 General-Purpose Timers

8.1 General-purpose timer 16-bit Edge Count or Edge Time mode does not load reload value

Description:

In Edge Count or Edge Time mode, the input events on the CCP pin decrement the counter until the count matches what is in the **GPTM Timern Match (GPTMTnMATCHR)** register. At that point, an interrupt is asserted and then the counter should be reloaded with the original value and counting begins again. However, the reload value is not reloaded into the timer.

Workaround:

Rewrite the **GPTM Timern Interval Load (GPTMTnILR)** register before restarting.

Silicon Revision Affected:

A0

8.2 The General-Purpose Timer match register does not function correctly in 32-bit mode

Description:

The **GPTM Timer A Match (GPTMTAMATCHR)** register triggers a match interrupt and a DMA request, if enabled, when the lower 16 bits match, regardless of the value of the upper 16 bits.

Workaround:

None.

Silicon Revision Affected:

A0

8.3 A spurious DMA request is generated when the timer rolls over in Input-Edge Time mode

Description:

When the timer is in Input-Edge Time mode and rolls over after the terminal count, a spurious DMA request is generated.

Workaround:

Either ignore the spurious interrupt, or disable DMA requests shortly before the terminal count and enable them again shortly after the terminal count.

Silicon Revision Affected:

A0

9 ADC

9.1 Use of "Always" triggering for ADC Sample Sequencer 3 does not work

Description:

When using ADC Sample Sequencer 3 (SS3) and configuring the trigger source to "Always" to enable continuous sampling by programming the SS3 Trigger Select field (EM3) in the **ADC Event Multiplexer Select (ADCEMUX)** register to 0xF, the first sample will be captured, but no further samples will be updated to the sequencer FIFO. Interrupts are continuously generated after the first sample and the FIFO status remains empty.

Workaround:

Software must disable and re-enable the sample sequencer to capture another sample.

Silicon Revision Affected:

A0

9.2 Incorrect behavior with timer ADC triggering when another timer is used in 32-bit mode

Description:

When a timer is configured to trigger the ADC and another timer is configured to be a 32-bit periodic or one-shot timer, the ADC is triggered continuously instead of the specified interval.

Workaround:

Do not use a 32-bit periodic or one-shot timer when triggering ADC. If the timer is in 16-bit mode, the ADC trigger works as expected.

Silicon Revision Affected:

A0

9.3 ADC hardware averaging produces erroneous results in differential mode

Description:

The implementation of the ADC averaging circuit does not work correctly when the ADC is sampling in differential mode and the difference between the voltages is approximately 0.0V.

Workaround:

Do not use hardware averaging in differential mode. Instead, use the FIFO to store results and average them in software.

Silicon Revision Affected:

A0

9.4 Retriggering a sample sequencer before it has completed the current sequence results in continuous sampling

Description:

Re-triggering a sample sequencer before it has completed its programmed conversion sequence causes the sample sequencer to continuously sample. If interrupts have been enabled, interrupts are generated at the appropriate place in the sample sequence. This problem only occurs when the new trigger is the same type as the current trigger.

Workaround:

Ensure that a sample sequence has completed before triggering a new sequence using the same type of trigger.

Silicon Revision Affected:

A0

9.5 ADC inputs have the incorrect channel assignments

Description:

Instead of channels 0, 1, 2, 3, 4, and 5 available on pins PE3, PE2, PE1, PE0, PD3, and PD2 respectively, the available channels are 4, 5, 6, 7, 0 and 1 on pins PD3, PD2, PD1, PD0, PE3, and PE2 respectively.

Workaround:

Software must utilize the alternate channels when programming the ADC (for example, use ADC4 instead of ADC0). When configuring the **GPIOAMSEL** register, software must enable analog inputs for the alternate pins (for example, PD3 instead of PE3).

Silicon Revision Affected:

A0

Fixed:

Fixed devices have 8 channels available so that board changes are not required and have date codes of 0x07 or later.

10 UART

10.1 The RTRIS bit in the UARTRIS register is only set when the interrupt is enabled

Description:

The RTRIS (UART Receive Time-Out Raw Interrupt Status) bit in the **UART Raw Interrupt Status (UARTRIS)** register should be set when a receive time-out occurs, regardless of the state of the enable RTIM bit in the **UART Interrupt Mask (UARTIM)** register. However, currently the RTIM bit must be set in order for the RTRIS bit to be set when a receive time-out occurs.

Workaround:

For applications that require polled operation, the RTIM bit can be set while the UART interrupt is disabled in the NVIC using the IntDisable(n) function in the StellarisWare Peripheral Driver Library, where n is 21, 22, or 49 depending whether UART0, UART1 or UART2 is used. With this configuration, software can poll the RTRIS bit, but the interrupt is not reported to the NVIC.

Silicon Revision Affected:

A0

11 I2C

11.1 I²C Slave Masked Interrupt Status (I2CSMIS) register bits are incorrect

Description:

The bits within the **I²C Slave Masked Interrupt Status (I2CSMIS)** register do not operate properly. This register should not be used by software.

Workaround:

Software should use the **I²C Slave Raw Interrupt Status (I2CSRIS)** register instead.

Silicon Revision Affected:

A0

12 USB

12.1 Device Capabilities 6 (DC6) register incorrectly specifies USB OTG functionality

Description:

On the LM3S5732 controller, the **Device Capabilities 6 (DC6)** register incorrectly indicates that the part has OTG capability when it provides Host or Device connectivity (bits 1:0 read 0x3 when they should read 0x2).

Workaround:

Do not attempt to use USB OTG functionality on this part. USB Host and Device capabilities are unaffected.

Silicon Revision Affected:

A0

12.2 Violating USB VBUS in-rush current specifications may cause USB controller to hang

Description:

Care should be taken to ensure that third-party USB devices used do not violate the USB specifications for maximum in-rush VBUS current. Connecting an out-of-spec USB device to the Stellaris USB controller may cause undesirable voltage glitches on the VBUS pin resulting in a controller hung condition.

Workaround:

In the event of a VBUS glitch or error on the PB1 GPIO, a VBUS error interrupt is generated, so software is capable of detecting the glitch. As a guideline, the following strategy is suggested:

1. Disable the external VBUS regulator.
2. Reset the USB controller (for example, with the Stellaris Peripheral Driver Library function `SysCtlPeripheral Reset(SYSCTL_PERIPH_USB0);`).
3. Reinitializes the USB controller.
4. Enable the external VBUS regulator.

In experiments, success is most likely if steps 2 and 3 are as close to each other as possible.

Note: This errata affects the USB controller if GPIO PB1 is connected to VBUS rather than an independent 5-V supply. If an independent 5-V supply is used, the glitch is unlikely to affect the PB1 voltage.

Silicon Revision Affected:

A0

12.3 PB0 and PB1 pins may not be used for GPIO or peripheral functions if USB Host or Device mode functionality is used

Description:

If the LM3S5732 microcontroller is used as a USB Host or USB Device controller, then the PB0 and PB1 pins must be tied to appropriate voltage levels.

Workaround:

In order to use the USB controller in Host or Device modes, the PB1 pin must be tied to 5 V (4.75-5.25 V). In addition, the PB0 pin must be tied Low for USB Host operation or tied High for USB Device operation. Note that when tying PB0 low through a resistor, the value of the resistor should not exceed the $R_{A\_PLUG\_ID}$ specification of 10 Ω for the USB cable. The PB0 and PB1 pins cannot be used for GPIO or peripheral functions if USB functionality is used.

Silicon Revision Affected:

A0

12.4 Transfer may stall when size of µDMA transfer does not match USB FIFO

Description:

When µDMA is used with USB to transfer data to or from the USB FIFO, and the size of the µDMA transfer does not match the size of the USB FIFO, the transfer can be stalled.

Workaround:

The µDMA channel should be configured with an arbitration size that matches the size of the USB FIFO. The size of the µDMA transfer should be restricted to whole multiples of the size of the USB FIFO. This applies for both read and write transfers of the USB FIFOs using µDMA.

For example, the USB endpoint is configured with a FIFO size of 64 bytes. The µDMA channel should be configured with an arbitration size of 64. The µDMA channel can be used to transfer 64 bytes to or from the endpoint FIFO. If the number of bytes to transfer is less than 64, then a programmed I/O method should be used to copy the data to or from the FIFO.

Silicon Revision Affected:

A0

12.5 USB Host controller may not be used to communicate with a low-speed Device when connected through a hub

Description:

Occasionally when the USB controller is operating as a Host and a low-speed packet is sent to a Device when connected through a hub, the subsequent Start-of-Frame will be corrupted. After a period of time, this corruption causes the USB controller to lose synchronization with the hub, resulting in data corruption.

Workaround:

None.

Silicon Revision Affected:

A0

Fixed:

Fixed on devices with date codes of 0x1A or later. In addition, the system clock on the MCU must be at least 30 MHz.

12.6 The USB PLL may fail to lock after reset

Description:

The USB PLL may fail to lock after any type of reset approximately 1 in every 100 times. Once the lock failure has occurred, a software reset or core reset does not correct it. An external reset or a power-on reset is required to correct the condition. As a result of the lock failure, the USB controller fails to enumerate onto the USB bus. Failure to enumerate can be detected by reading the values of bits [4:3] in the **USB Device Control (USBDEVCTL)** register. If both of these bits are 0, the device has failed to enumerate.

Workaround:

There is a workaround for software reset. There are two types of software reset, SYSRESREQ and VECTRESET. SYSRESREQ resets the entire device, whereas VECTRESET only resets the core. This problem can be eliminated by performing a VECTRESET using this code:

```
HWREG(NVIC_APINT) = NVIC_APINT_VECTKEY | NVIC_APINT_VECT_RESET;
```

VECTRESET only resets the core, so if any peripherals must be reset, use the peripheral software reset function, SysCtlPeripheralReset(). Note that if the USB module is reset using SysCtlPeripheralReset(), the USB PLL may fail to lock.

For any other type of reset, after detecting the enumeration failure, use a GPIO or an external watchdog device to issue an external reset.

Silicon Revision Affected:

A0

Fixed:

Not yet fixed.

13 Electrical Characteristics

13.1 Certain pins do not fully comply with the JEDEC ESD standard

Description:

The pins listed below do not fully meet the industry ESD standard of 2.0 KV Human Body Model (HBM) under all conditions. All Stellaris devices are tested using JEDEC Standard JESD22-A114. These pins fail only in the pin-to-power condition using a positive voltage transient. The pins pass at 500 V. All other HBM conditions pass on these pins. All other device pins fully comply with this HBM JEDEC standard.

- PB0
- PB1
- USB0DM
- USB0DP
- USB0RBIAS
- PF0
- OSC0
- OSC1

Workaround:

Extra caution should be taken to ensure proper ESD handling of these devices. Use appropriate caution when implementing ESD protection circuits on signals routed to these pins.

Silicon Revision Affected:

A0

Texas Instruments Incorporated
108 Wild Basin, Suite 350
Austin, TX 78746
http://www.ti.com/stellaris
http://www-k.ext.ti.com/sc/technical-support/product-information-centers.htm

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Mobile Processors	www.ti.com/omap		
Wireless Connectivity	www.ti.com/wirelessconnectivity		
	TI E2E Community Home Page		e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265