Errata

MSPM0L130x-Q1 Microcontrollers

ABSTRACT

This document describes the known exceptions to the functional specifications (advisories).

Table of Contents

1 Functional Advisories

Advisories that affect the device's operation, function, or parametrics.

✓ The check mark indicates that the issue is present in the specified revision.

Errata Number	Rev C
ADC_ERR_01	✓
ADC_ERR_02	✓
COMP_ERR_01	✓
GPIO_ERR_01	✓
I2C_ERR_01	✓
IO_ERR_01	✓
PMCU_ERR_01	✓
PMCU_ERR_02	✓
PMCU_ERR_03	✓
PWREN_ERR_01	✓
SPI_ERR_01	✓
SYSOSC_ERR_01	✓

2 Preprogrammed Software Advisories

Advisories that affect factory-programmed software.

✓ The check mark indicates that the issue is present in the specified revision.

3 Debug Only Advisories

Advisories that affect only debug operation.

✓ The check mark indicates that the issue is present in the specified revision.

4 Fixed by Compiler Advisories

Advisories that are resolved by compiler workaround. Refer to each advisory for the IDE and compiler versions with a workaround.

✓ The check mark indicates that the issue is present in the specified revision.

5 Device Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all MSP MCU devices. Each MSP MCU automotive family member has one of two prefixes: M0 or XM0. These prefixes represent evolutionary stages of product development from engineering prototypes (XM0) through fully qualified production devices (M0).

XM0 – Experimental device that is not necessarily representative of the final device's electrical specifications

M0 – Fully qualified production device

Support tool naming prefixes:

X: Development-support product that has not yet completed Texas Instruments internal qualification testing.

null: Fully-qualified development-support product.

XM0 devices and X development-support tools are shipped against the following disclaimer:

"Developmental product is intended for internal evaluation purposes."

MSP devices have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

Predictions show that prototype devices (X) have a greater failure rate than the standard production devices. TI recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the temperature range, package type, and distribution format.

5.1 Device Symbolization and Revision Identification

Table 5-1 below defines the device revision to version ID mapping.

Table 5-1. Die Revisions

Revision Letter (package marking)	Version (in the device factory constants memory)
C	1

The revision letter indicates the product hardware revision. Advisories in this document are marked as applicable or not applicable for a given device based on the revision letter. This letter maps to an integer stored in the memory of the device, which can be used to look up the revision using application software or a connected debug probe.

6 Advisory Descriptions

ADC_ERR_01 *ADC is unable to trigger fast clock in STANDBY1 mode*

Revisions Affected

Rev C

Details

When the device is operating in STANDBY1 mode, the ADC module may not correctly assert an asynchronous fast clock request when it is triggered through the event system (for example, when an event publisher such as a timer generates an event to the ADC through the ADC event subscriber port).

Workaround

Use STANDBY0 or higher power modes when triggering ADC conversions via the event fabric.

ADC_ERR_02 *ADC does not release the fast clock request in between periodical triggers*

Revisions Affected

Rev C

Details

When the ADC is set up in repeat mode and triggered periodically via the event fabric, it does not release its fast clock request in between triggers. IF so, the ADC will hold the clock request AND consume extra power.

Workaround

Configuring the ADC in single mode (run to completion with ADC disabling at end of channel/sequence), AND THEN using a software interrupt at the end of the ADC sequence to re-enable the ADC to wait for the next trigger.

COMP_ERR_01 *Comparator output keeps toggling in STANDBY0 mode*

Revisions Affected

Rev C

Details

When the comparator inverting input mux (IMSEL) is set to 0 (COMP0_IN0-), and the device operating mode is set to STANDBY0, the comparator output may toggle unexpectedly regardless of the applied input voltages to the comparator.

Workaround

If utilizing comparator in STANDBY0, and applying an external voltage to the negative terminal, utilize channel 1 of the comparator (IMSEL = 1).

If utilizing comparator in STANDBY0, and applying an internal reference voltage (DAC8, VREF, etc.), set IMSEL to a value other than 0 (IMSEL ≠ 0).

GPIO_ERR_01 *GPIO wakeup edges may be lost in STANDBY mode*

Revisions Affected

Rev C

Details

After waking up once through a single GPIO edge, subsequent GPIO wakeup edges can be missed in STANDBY modes.

- Case 1: STANDBY0 wakeup - IF the MCU is set into STANDBY0 mode before one cycle of LFCLK AND you set the IO back to the “non-wake” state for <1 LFCLK cycle AND THEN assert it again, the next wakeup edge will not be detected. This issue is because the GPIO module never sees the IO "clear" to the "non-asserted" state AND so it thinks that the pin was always asserted.
- Case 2: STANDBY1 wakeup - IF a GPIO edge is used to wakeup AND the GPIO pulse is still active when the device returns to STANDBY1 THEN the device will not detect any subsequent wakeup edges.

Workaround

- Case 1:
  - Set GPIO wakeup edge to both falling and rising edges
  - OR Ensure GPIO wakeup pulse is longer than one LFCLK cycle
- Case 2:
  - Set GPIO wakeup edge to both falling and rising edges
  - OR Ensure GPIO wakeup pulse is not active before entering STANDBY1

I2C_ERR_01 *I2C module may hold the SDA line in SBMUS mode when an SMBUS quick command is issued*

Revisions Affected

Rev C

Details

When the I2C module is target mode and configured for SBMUS, IF the bus controller issues an SMBUS quick command addressed to the device (an I2C START condition followed by a 7-bit address, 1-bit R/W signal, 1-bit ACK, and an I2C STOP condition) with the R/W bit set to read, THEN the I2C module may attempt to pull the SDA line low at the same time that the bus controller is attempting to signal the I2C STOP condition, preventing the STOP condition from completing successfully.

Workaround

Load data into the I2C module transmit FIFO with the MSB set to 1 before the address ACK is completed to prevent the I2C module from driving the SDA line low. This will allow the bus controller to issue the STOP condition successfully and complete the SMBUS quick command.

IO_ERR_01 *High current draw following a negative current injection on PA2 pin*

Revisions Affected

Rev C

Details

A negative current injection into the PA2 pin can lead to an unexpected and sustained high current draw in the device.

Workaround 1

If the PA2 is used in ROSC mode (100kΩ resistor is populated between ROSC and VSS), no additional handling is required.

Workaround 2

Place a series resistor with a typical value of 100 Ω between the PA2 pin and any connected circuit, with the resistor placed near to the PA2 pin. This resistor is intended to limit fast transients seen at the PA2 pin and is sufficient to prevent this scenario from occurring.

Workaround 3

Avoid using the PA2 pin and use an alternate pin instead.

PMCU_ERR_01 *Current leakage on PA2/ROSC pin*

Revisions Affected

Rev C

Details

When using pin PA2/ROSC in a functional configuration that is not ROSC, a large leakage current can be observed if the pin is driven high. This is due an unintended connection to ground through an impedance of approximately 17k ohm.

This errata does not affect ROSC functionality.

Workaround

IF PA2/ROSC is used for alternative functionality to ROSC, THEN one must account for the extra leakage current in energy budgets when the pin is driven high, either externally or internally. Also, be mindful of potential voltage divider creation if the pin is pulled up externally with a resistor.

PMCU_ERR_02 *Transient voltage output during the device startup*

Revisions Affected

Rev C

Details

When using pin PA2/ROSC in a functional configuration that is not ROSC, the observed phenomenon of a transient voltage output during the device startup is caused by an unintended operation on an internal switch, resulting in a lower impedance path from VCORE to ground. This errata does not affect ROSC functionality.

Workaround

IF PA2/ROSC is used for alternative functionality to ROSC, any pulses generated through PA2 during the startup phrase should be ignored.

PMCU_ERR_03 *BOR1, BOR2, and BOR3 do not work in STANDBY mode*

Revisions Affected

Rev C

Details

The user-selectable alternate BOR thresholds (BOR1, BOR2, BOR3) are not functional when operating in the device in STANDBY mode.

Workaround

Do not use BOR1, BOR2, or BOR3 when operating in STANDBY mode. Configure the device to use the default BOR0 level before entering STANDBY mode. Upon exit from STANDBY mode, alternate BOR levels can be re-enabled.

PWREN_ERR_01 *Peripheral registers are still accessible after disabling PWREN register*

Revisions Affected

Rev C

Details

When disabling the power of a peripheral by setting the PWREN register to 0, the peripheral's registers may appear to retain data values if read. Reading or writing to the registers when PWREN is 0 has no affect as the peripheral has no effect.

The following peripherals are affected: comparator (COMP), operational amplifier (OPA), TimerA, TimerG, general-purpose input/output (GPIO), and windowed watchdog timer (WWDT).

Workaround

When the PWREN register of the peripheral is set to 0, the values of the associated registers should be disregarded or considered invalid.

SPI_ERR_01 *SPI Parity bit is not functional*

Revisions Affected

Rev B

Details

SPI hardware parity modes are not functional.

Workaround

Do not enable hardware parity via the PTEN or PREN bit in the CTL1 register of the SPI module. Parity computation and checking may be implemented with application software.

SYSOSC_ERR_01 ***MFCLK drift when using SYSOSC FCL together with STOP1 mode***

Revisions Affected

Rev C

Details

When MFCLK is enabled AND SYSOSC is using the frequency correction loop (FCL) mode AND the STOP1 low power operating mode is used, THEN the MFCLK may drift by 2 cycles when SYSOSC shifts from 4MHz back to 32MHz (either upon exit from STOP1 to RUN mode or upon an asynchronous fast clock request that forces SYSOSC to 32MHz).

Workaround1

Use STOP0 mode instead of STOP1 mode. There is no MFCLK drift when STOP0 mode is used.

Workaround2

Do not use SYSOSC in the FCL mode (leave FCL disabled) when using STOP1.

7 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

DATE	REVISION	NOTES
December 2023	*	Initial Release

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

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