MSP430F23x0 Device Erratasheet

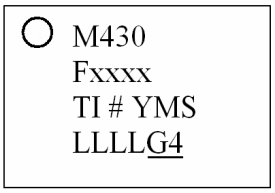
Current Version

Devices	Rev:	CPU14	FLASH22	JTAG14	PORT10	TA12	TA16	TB2	TB16	USCI15	USCI16	USCI17	USCI18	XOSC5
MSP430F2330	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2350	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2370	C	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: See Appendix for prior revisions

Package Markings

RHA40: QFN (RHA) 40-pin



YM = Year and Month Date Code
LLLL = LOT Trace Code
S = Assembly Site Code
# = DIE Revision
o = PIN 1

Detailed Bug Description

CPU14 CPU14 - Bug description:

Module: CPU, Function: Erroneous setting of SCG0 after reset

The SCG0 bit in the CPU status register (SR) is set after any reset (PUC or POR) if bit #6 in the reset vector destination address is set. Setting SCG0 turns off the DCO DC generator when DCOCLK is not used for MCLK or SMCLK.

Workarounds:
1) As the error only occurs after PUC or POR, it is sufficient to clear the SCG0 bit at the beginning of the program code. Example:
   bic.w #SCG0, SR
2) Avoid using reset destination addresses where bit #6 is set. Allowed reset vector destination addresses are: xx0xh, xx1xh, xx2xh, xx3xh, xx8xh, xx9xh, xxAxh, xxBxh.

When any of the above reset destination addresses are used, SCG0 will be valid.

FLASH22 FLASH22 - Bug description:

Module: Flash, Function: Flash controller may prevent correct LPM entry

In case ACLK (or SMCLK) is used as the Flash controller clock source, and this clock source gets deactivated due to a low-power mode entry while a Flash erase or write operating is pending, the Flash controller will keep ACLK (or SMCLK) active even after the Flash operation has been completed. This will result in an incorrect LPM entry and increased current consumption. Note that this issue can only occur in case the Flash operation and the low-power mode entry are initiated from code located in RAM.

Workaround:
Do not enter low-power modes while Flash erase or write operations are active. Wait for the operation to be completed before entering a low-power mode.

JTAG14 JTAG14 - Bug description:

Module EEM: Releasing JTAG control can corrupt CPU registers during debug

During a debug session, on rare occasions, the CPU register contents can get corrupted when JTAG control is released by the debugger. This behavior is exhibited during, but not limited to, the use of the “Use Virtual Breakpoints” and “Force Single Stepping” features in the IAR Embedded Workbench software. This bug does not affect normal device and application operation, such as starting a device out of POR and executing application code.

In order for the bug to occur, both of the following two conditions must be true:

1 – The CPU (MCLK) is sourced by the DCO
AND
2 – The “External Resistor (Rosc)” feature of the DCO is not used

Workaround:
Use an external crystal or a digital high-speed clock source connected to the LFXT1 oscillator to source the CPU (MCLK) during a debug session. Alternatively, use the on-chip DCO in the “External Resistor (Rosc)” configuration. Note that in this case an external resistor connected to the device’s Rosc-pin is mandatory, and that the factory-programmed DCO calibration constants cannot be applied directly.

Detailed Bug Description (continued)

PORT10 Port10 - Bug description:

Module: Digital I/O, Function: Pullup/pulldown resistor selection when module pin function is selected

When the pullup/pulldown resistor for a certain port pin is enabled (PxREN.y=1) and the module port pin function is selected (PxSEL.y=1), the pullup/pulldown resistor configuration of this pin is controlled by the respective module output signal (Module X OUT) instead of the port output register (PxOUT.y).

Workaround:
None. Do not set PxSEL.y and PxREN.y at the same time.

TA12 TA12 - Bug description:

Module: TimerA, Function: Interrupt is lost (slow ACLK)

TimerA counter is running with slow clock (external TACLK or ACLK) compared to MCLK. The compare mode is selected for the capture/compare channel and the CCRx register is incremented by 1 with the occurring compare interrupt (if TAR = CCRx).
Due to the fast MCLK, the CCRx register increment (CCRx = CCRx+1) happens before the TimerA counter has incremented again. Therefore the next compare interrupt should happen at once with the next TimerA counter increment (if TAR = CCRx + 1). This interrupt gets lost.

Workaround:
Switch capture/compare mode to capture mode before the CCRx register increment. Switch back to compare mode afterwards.

TA16 TA16 - Bug description:

Module: TimerA, Function: First increment of TAR erroneous when IDx > 00

The first increment of TAR after any timer clear event (POR/TACLR) happens immediately following the first positive edge of the selected clock source (INCLK, SMCLK, ACLK or TACLK). This is independent of the clock input divider settings (ID0, ID1). All following TAR increments are performed correctly with the selected IDx settings.

Workaround:
None

Detailed Bug Description (continued)

TB2 TB2 - Bug description:

Module: TimerB, Interrupt is lost (slow ACLK)

TimerB counter is running with slow clock (external TBCLK or ACLK) compared to MCLK. The compare mode is selected for the capture/compare channel and the CCRx register is incremented by 1 with the occurring compare interrupt (if TBR = CCRx).
Due to the fast MCLK, the CCRx register increment (CCRx = CCRx+1) happens before the TimerB counter has incremented again. Therefore the next compare interrupt should happen at once with the next TimerB counter increment (if TBR = CCRx + 1). This interrupt gets lost.

Workaround:
Switch capture/compare mode to capture mode before the CCRx register increment. Switch back to compare mode afterward.

TB16 TB16 - Bug description:

Module: TimerB, Function: First increment of TBR erroneous when IDx > 00

The first increment of TBR after any timer clear event (POR/TBCLR) happens immediately following the first positive edge of the selected clock source (INCLK, SMCLK, ACLK or TBCLK). This is independent of the clock input divider settings (ID0, ID1). All following TBR increments are performed correctly with the selected IDx settings.

Workaround:
None

USCI15 USCI15 - Bug description:

Module: USCI, Function: Receive buffer overrun undetected

When a new character is being loaded into RXBUF by the USCI hardware, the previous character may be overwritten. This can occur when the USCI hardware updates RXBUF with the new character and in the same instant the CPU accesses RXBUF to read the old character. In this case, the old character is lost and the new one is read out. No receive overrun error will be detected and UCOE will not be set.

Workaround:
Running the CPU at an adequate speed in order to guarantee access of RXBUF of received characters prior to new character receive completion will minimize the potential that simultaneous access of RXBUF may happen.

Detailed Bug Description (continued)

USCI16 USCI16 - Bug description:

Module: USCI, Function: UART/IrDA Mode Lost Characters

When configured for UART/IrDA mode, the USCI baud rate generator may halt operation under the following conditions:
1 - IrDA mode: repeated invalid start bits on the receive line
or
2 - UART/IrDA modes: positive pulse on the receive line during break character reception inside the stop bit time slot (the second stop bit time slot in case of UCSPB=1) with a pulse width that passes the deglitch filter but is shorter than half a bit time.

After halting, additional characters will be ignored. Transmit functionality is not affected.

Workaround:
Check the UCBUSY flag status periodically in software. If the flag is set and no character has been received in the expected time, reset the USCI module in software. To reset the USCI module, toggle UCSWRST and re-enable the USCI interrupts.

USCI17 USCI17 - Bug description:

Module: USCI, Function: UCSTOE flag set too early in automatic baud rate detection mode

When configured for automatic baud rate detection UART mode (UCMODEx=11), the synch field timeout flag (UCSTOE) is already set after 0x3FFFh BRCLK cycles instead of 0x7FFF as documented in the device user's guide.

Workaround:
Use a lower BRCLK frequency if needed for the baud rate to be detected.

USCI18 USCI18 - Bug description:

Module: USCI, Function: Framing error detected as break in automatic baud rate mode

When configured for automatic baud rate detection UART mode (UCMODEx=11), a received byte with a framing error can be detected as a break.

Workaround:
Switch to UART mode (UCMODEx=00) after auto baud rate detection.

XOSC5 XOSC5 - Bug description:

Module: LFXT1 OSC: LF crystal failures may not be properly detected by the oscillator fault circuitry

The oscillator fault error detection of the LFXT1 oscillator in low frequency mode (XTS = 0) may not work reliably causing a failing crystal to go undetected by the CPU; i.e., OFIFG will not be set.

Workaround:
None

Appendix: Prior Versions

None

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 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.

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.

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

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
Low Power Wireless	www.ti.com/lpw	Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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