MSP430F261x/241x Device Erratasheet

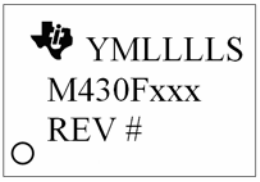
Current Version

Devices	Rev:	BCL12	CPU7	CPU8	DMA3	DMA4	FLASH24	FLASH25	TA12	TA16	TB2	TB16	USCI20	USCI21	XOSC5
MSP430F2416	E	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2417	E	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2418	E	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2419	E	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2616	E	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2617	E	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2618	E	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2619	E	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: See Appendix for prior revisions

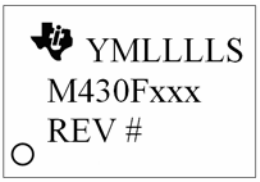
Package Markings

PM64: LQFP(PM) 64-pin



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

PN80: LQFP(PN) 80-pin



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

Detailed Bug Description

BCL12 BCL12 - Bug description

Module: BasicClock, Function: Switching RSEL can cause DCO dead time

After switching RSELx bits (located in register BCSCTL1) from a value of >13 to a value of <12, the resulting clock delivered by the DCO can stop for a certain time before the new clock frequency gets applied. This dead time is in the range of 20 µs.

Workaround:
Use an intermediate step when switching RSEL from >13 to <12. The intermediate RSEL value should be 13.

CURRENT RSEL	TARGET RSEL	RECOMMENDED TRANSITION SEQUENCE
15	14	Switch directly to target RSEL
14 or 15	13	Switch directly to target RSEL
14 or 15	0 to 12	Switch to 13 first, and then to target RSEL (two step sequence)
0 to 13	0 to 12	Switch directly to target RSEL

CPU7 CPU7 - Bug description

Module: CPU, Function: CPU CALL and PUSH instruction

CALL and PUSH instructions using @SP+ , @SP, and X(SP) addressing modes behave differently from the original CPU. The original CPU predecremented the SP, then used the predecremented value to calculate the address. The new CPU uses the SP to calculate the address, then decrements it.

Workaround: (Fixed in the compiler for C source. No silicon fix planned.)
None.

CPU8 CPU8 - Bug description

Module: CPU, Function: Using odd values in the SP register

The SP can be written with odd values. In the original CPU, an odd SP value could be combined with an odd offset [i.e., mov. #value, 5(SP)]. In the new CPU, the SP can be written with an odd value, but the first time the SP is used, the LSB is forced to 0.

Workaround:
Do not use odd values with the SP.

Detailed Bug Description (continued)

DMA3 DMA3 - Bug description

Module: DMA, Function: Read-modify-write instructions may corrupt DMA address registers

When a 16-bit-wide read-modify-write instruction (such as add.w and sub.w) is directly used on a DMA address register (DMAxSA or DMAxDA), the register contents will get corrupted.

Workaround:
1) Do not use 16-bit-wide read-modify-write instructions on DMA address registers. Instead, in case address calculations are necessary, do the calculations first, and then assign the result to the DMA address registers.

or

2) Use 20-bit-wide read-modify-write instructions (such as addx.a, subx.a) on the DMA address registers, if needed.

DMA4 DMA4 - Bug description

Module: DMA, Function: Corrupted write access to 20-bit DMA registers

When a 20-bit-wide write to a DMA address register (DMAxSA or DMAxDA) is interrupted by a DMA transfer, the register contents may be unpredictable.

Workaround:
1) Ensure that no DMA access interrupts 20-bit-wide accesses to the DMA address registers. This can be achieved by temporarily disabling all active DMA channels (DMAEN = 0), or by enabling the DMA on fetch feature (DMAONFETCH = 1). Also, it may be possible to ensure this by system design.

or

2) Use word access for accessing the DMA address registers. Note that this limits the address values that can be written to word values.

FLASH24 FLASH24 - Bug description

Module: Flash, Function: Write or erase emergency exit can cause failures

When a flash write or erase is abruptly terminated, any further reliable reads by the flash controller are not ensured. The abrupt termination can be the result of one the following events:

1) The flash controller clock is configured to be SMCLK sourced by an external crystal. An oscillator fault occurs, thus stopping this clock abruptly.

or

2) The Emergency Exit bit (EMEX in FCTL3), when set, forces a write or an erase operation to be terminated before normal completion.

or

3) The Enable Emergency Interrupt Exit bit (EEIEX in FCTL1), when set with GIE = 1, can lead to an interrupt causing an emergency exit during a flash operation.

Workaround:
1) Do not use SMCLK as the source for the flash controller clock if it is sourced from an external crystal.
2) After setting EMEX = 1, wait for a sufficient amount of time before flash is accessed again.
3) No workaround. Do not use EEIEX bit.

Detailed Bug Description (continued)

FLASH25 FLASH25 - Bug description

Module: Flash, Function: Marginal read mode is not functional

The control bits for marginal read mode contained in the FCTL4 register are automatically cleared by any flash access. This prevents the marginal read mode from being used.

Workarounds:

It is possible to read out memory contents in marginal read mode if the indexed addressing mode X(Ry) is used to access the flash memory. In this case, the FCTL4 control bits will not get cleared, and the marginal read mode works as expected. It is recommended to write the code for reading the flash memory contents in assembler, as this allows full control over the addressing mode that is used. Note that certain assemblers may optimize an indexed addressing source operation of 0(Ry) to an indirect register mode @Ry operation, which will not work. The following is an example of reading the word memory location 0x4000 in marginal read mode, preventing a possible assembler optimization:

```
mov.w #0x4000,R15 ; Pointer to target address
dec.w R15         ; Decrement pointer
mov.w 1(R15),R12  ; Read memory contents at R15 + 1, store result in R12
```

TA12 TA12 - Bug description

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

Timer_A 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 Timer_A counter has incremented again. Therefore, the next compare interrupt should happen at once with the next Timer_A 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: Timer_A, 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: Timer_B, Function: Interrupt is lost (slow ACLK)

Timer_B 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 Timer_B counter has incremented again. Therefore, the next compare interrupt should happen at once with the next Timer_B 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 afterwards.

TB16 TB16 - Bug description

Module: Timer_B, 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

Detailed Bug Description (continued)

USCI20 USCI20 - Bug description

Module: USCI, Function: I2C mode multi-master transmitter issue

When configured for I2C master-transmitter mode and used in a multi-master environment, the USCI module can cause unpredictable bus behavior if all of the following four conditions are true:

1) Two masters are generating SCL.
and
2) The slave is stretching the SCL low phase of an ACK period while outputting NACK on SDA.
and
3) The slave drives ACK on SDA after the USCI has already released SCL, and then the SCL bus line gets released.
and
4) The transmit buffer has not been loaded before the other master continues communication by driving SCL low.

The USCI will remain in the SCL high phase until the transmit buffer is written. After the transmit buffer has been written, the USCI will interfere with the current bus activity and may cause unpredictable bus behavior.

Workaround:
1) Ensure that the slave doesn't stretch the SCL low phase of an ACK period.
or
2) Ensure that the transmit buffer is loaded in time.
or
3) Do not use the multi-master transmitter mode.

USCI21 USCI21 - Bug description

Module: USCI, Function: UART IrDA receiver mode

IrDA reception does not function correctly at certain baud rates. This occurs only when the IrDA receive filter via the UCAxIRRCTL register is enabled and the USCI source clock (BRCLK) is higher than 6 MHz.

Workaround:
1) Set the filter length (UCIRRXFLx in UCAxIRRCTL) to the maximum value 0x3F to achieve proper functionality.
2) Reduce the frequency of the USCI source clock (BRCLK) to the IrDA module to be less than 6 MHz.

XOSC5 XOSC5 - Bug description

Module: LFXT1 OSC, Function: 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

Devices	Rev:	ADC18	ADC19	BCL12	COMP2	CPU7	CPU8	DMA3	DMA4	FLASH22	FLASH23	FLASH24	FLASH25	PORT10	TA12	TA16	TB2	TB16	TB19	USCI16	USCI20	USCI21	XOSC5	XOSC6
MSP430F2416	E			✓		✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	D			✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	A	✓		✓	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2417	E			✓		✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	D			✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	A	✓		✓	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2418	E			✓		✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	D			✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	A	✓		✓	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2419	E			✓		✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	D			✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓					✓	✓		✓	✓	✓	✓			✓	✓	✓	
	A	✓		✓	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2616	E			✓		✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	D			✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	A	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2617	E			✓		✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	D			✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	A	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2618	E			✓		✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	D			✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	A	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430F2619	E			✓		✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
MSP430F2619	D			✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
	B		✓	✓	✓	✓	✓	✓	✓			✓	✓		✓	✓	✓	✓			✓	✓	✓	
MSP430F2619	A	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Detailed Bug Description

ADC18 ADC18 - Bug description

Module: ADC12, Function: Incorrect conversion result in extended sample mode

The ADC12 conversion result can be incorrect in case where the extended sample mode is selected (SHP = 0), the conversion clock is not the internal ADC12 oscillator (ADC12SSEL > 0), and one of the following two conditions is true:

1) The extended sample input signal SHI is asynchronous to the clock source used for ADC12CLK, and the undivided ADC12 input clock frequency exceeds 3.15 MHz.

or

2) The extended sample input signal SHI is synchronous to the clock source used for ADC12CLK, and the undivided ADC12 input clock frequency exceeds 6.3 MHz.

Workaround:

1) Use the pulse sample mode (SHP = 1).

or

2) Use the ADC12 internal oscillator as the ADC12 clock source.

or

3) Limit the undivided ADC12 input clock frequency to 3.15 MHz.

or

4) Use the same clock source (such as ACLK or SMCLK) to derive both SHI and ADC12CLK in order to achieve synchronous operation, and also limit the undivided ADC12 input clock frequency to 6.3 MHz.

ADC19 ADC19 - Bug description

Module: ADC12, Function: Sample start in extended pulse mode

When operating in extended pulse mode, if ADC12SC is set in the same instruction as the first setting of ENC, a sample will not be started (that is, the ADC12SC bit will have no effect). Instead, ENC must be set at least one instruction prior to the first occurrence of ADC12SC being set.

Workaround:
Set ENC in a separate instruction prior to setting ADC12SC.

COMP2 COMP2 - Bug description

Module: Comparator_A+, Function: Configuring the port disable register (CAPD)

According to the user's guide, each bit in the CAPD register should correspond with its associated port I/O number. For example, when bit 0 of CAPD is set, the port disable function of pin Px.0 is enabled, bit 1 controls Px.1, and so on (where Px is the port that contains the comparator inputs). However, on this device, the bits of the CAPD register correspond with the Comparator_A input number. For example, bit 0 of CAPD controls the CA0 input, bit 1 controls CA1, etc. This difference matters when the port I/O number is not the same as the comparator input number.

If the wrong CAPD bit is set, the port I/O function for the wrong pin will be disabled. Also, the analog signal applied to the comparator input pin being used may cause a parasitic current to flow from V_{CC} to GND. (See the Comparator_A + chapter of the *MSP430x2xx Family User's Guide* for more information on CAPD.)

Detailed Bug Description (continued)

FLASH22 FLASH22 - Bug description

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

When 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 when 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.

FLASH23 FLASH23 - Bug description

Module: Flash, Function: Erasing flash memory

The option to erase all main memory segments (MERAS = 1, ERASE = 0) does not apply to the entire main memory area. Flash arrays below and above the 64k address boundary must be erased separately.

Workaround:
Erase each main memory segment separately.

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.

TB19 TB19 - Bug description

Module: Timer_B, Function: TBIFG/TBIV not updated after access to TBIV

After any access to TBIV, the TBIFG flag should automatically be cleared, and the TBIV value should be updated to reflect the new state of Timer_B interrupts. However, they are not updated.

Workaround:
None

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 reenable the USCI interrupts.

XOSC6 XOSC6 - Bug description

Module: XT2 OSC, Function: XT2 crystal failures may not be properly detected by the oscillator fault circuitry

The XT2OF flag should be set if the XT2 frequency falls below 30 kHz. If there is no oscillation at all, the flag will still operate properly. However, 0 kHz to 30 kHz produces an undefined state on XT2OF. When this occurs, OFIFG will not be set.

Workaround:
Do not depend on the fault detection circuitry to accurately detect all failures.

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**	
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
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
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
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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