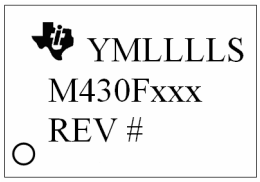
MSP430F543x Device Erratasheet (Experimental “X430” Si)
Current Version

Devices	Revision	ADC20	ADC21	CPU7	CPU15	CPU16	CPU17	DMA5	DMA6	DMAx	EEM4	EEM5	FLASH28	FLASHx	JTAG16	JTAG17	JTAGx	RTC1	PMM1	PMM2	PMM3	LPMx	SYS1	SYS2	TB20	TBx	UCS2	UCS3	USCI21
MSP430F5438 (“X430”)	I	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: See Appendix for prior revisions

Package Markings(1)

PZ100: LQFP(PZ) 100-pin



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

(1) M430Fxxxx is replaced by X430Fxxxx on prerelease experimental samples.

Detailed Bug Description

ADC20 ADC20 - Bug description

Module: ADC12, Function: ADC12DF does not work as described in the User Guide

The conversion data format (ADC12DF bit setting) must be selected before starting a conversion. Switching the conversion data format bit after a conversion has started does not deliver results in the newly selected format.

Workaround:
Change the conversion data format before starting the conversion.

ADC21 ADC21 - Bug description

Module: ADC12, Function: ADC12ENC is not working correctly

Resetting the ADC12ENC bit during an active conversion or within a sequence could result in a wrong state of the ADC12 module and undefined behavior during the next conversion.

Workaround:
Resetting the ADC12ENC bit during an active conversion or within a sequence should be avoided. If the bit is reset, the result should be discarded and a new conversion made.

CPU7 CPU7 - Bug description

Module: CPUX, Function: CPUXv2 CALL and PUSH instruction

CALL and PUSH instructions using SP behave differently from original CPU. Addressing modes affected when using CALL:
1) @SP
2) @SP+
3) X(SP)
Addressing modes affected when using PUSH:
1) @SP
2) @SP+
The original CPU pre-decrements the SP and then uses the pre-decremented value to calculate the address. The new CPUX and CPUXv2 use the SP to calculate the address and then decrement it. The instruction PUSH @SP+ in CPUXv2 and both instructions PUSH @SP+ and CALL @SP+ in CPUX do not auto-increment SP and, hence, behave similarly to the corresponding instructions in @SP addressing mode.

Workaround: (Fixed in the compiler for C source)
None

Detailed Bug Description (continued)

CPU15 CPU15 - 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 occurs only 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, and xxBxh.

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

CPU16 CPU16 - Bug description

Module: Core/CPU, Function: Indexed addressing with instructions calla, mova, and bra.

With indexed addressing mode and instructions calla, mova, and bra, it is not possible to reach memory above 64k if the register content is <64k.
Example: calla idx16 = 0004h(R5 = FFFEh) results in a 20-bit call of address 0002h instead of 10002h.

Workaround:
- Use different addressing mode to reach memory above 64k.
- Use adda index, Reg to calculate address in upper memory and then do a calla Reg.

CPU17 CPU17 - Bug description

Module: Core/CPU, Function: Consecutive interrupts (NMI after GMI)

If an NMI occurs within three system clock cycles after a GMI, the push of the PC/SP before service of the NMI is corrupted.

Workaround:
None

DMA5 DMA5 - 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.

Detailed Bug Description (continued)

DMA6 DMA6 - Bug description

Module: DMA, Function: DMA cannot write to DMA

One DMA channel cannot modify the registers of another DMA channel.

Workaround:
Modify DMA registers using CPU execution.

DMAx DMAx - Bug description

Module: DMA, Function: Single stepping in debug

When single stepping, the DMA does not always transfer the correct values.

Workaround:
None. Note: Occurs during debug; when executing at full speed, works as defined.

EEM4 EEM4 - Bug description

Module: Enhanced Emulation Logic

If the EEM is configured to generate DMA-dependent triggers by setting the TRIG3 bit in the Trigger Control Register of a memory bus trigger block, then a trigger could be missed or invalid trigger could be generated.

Workaround:
A trigger configuration can be used with TRIG3 bit set to 0, so trigger generation does not depend on DMA/non-DMA transactions. In this case, the triggers are generated correctly. The State Storage Block can be used afterward to determine whether the trigger causing transaction was a DMA or non-DMA transaction.

FLASH28 FLASH28 - Bug description

Module: Flash, Function: Read disturb

Flash access with default PMM settings (PMM level 0) or PMM level 1 is not reliable over the full device operating range and is not recommended.

Workaround:
Set PMM to V_{CORE} level three (1.75 V).

```
PMMCTL0_H = 0xA5;   // Open PMM module registers for write access
PMMCTL0 = 0xA500 + PMMCOREV_2;       // Set VCore to 1.75 V
SVSMLCTL = SVMLE + SVSMLRRL_6;       // Set SVM new Level
while ((PMMIFG & SVSMLDLYIFG) == 0); // Wait till SVM is settled
PMMIFG &= ~(SVMLVLRIFG + SVMLIFG);   // Clear already set flags
if ((PMMIFG & SVMLIFG))
  while ((PMMIFG & SVMLVLRIFG) == 0); // Wait till level is reached
SVSMLCTL &= ~SVMLE;                   // Disable Low side SVM
PMMCTL0_H = 0x00;   // Lock PMM module registers for write access
```

Detailed Bug Description (continued)

FLASHx FLASHx - Bug description

Module: Flash, Function: Page erase

When a flash operation is in progress, interrupts are not automatically disabled. The CPU always tries to service the interrupt request, whether or not the flash is busy.

Workaround:
When erasing flash, check the ACCVIFG (access violation flag) to ensure the operation was not interrupted. If it was, then redo the erase.

JTAG16 JTAG16 - Bug description

Module: JTAG, Function: JTAG mailbox is not working in all circumstances

It is not possible to load or execute the JTAG mailbox instruction while the RST/NMI pin is held low (reset active).

Workaround:
None

JTAG17 JTAG17 - Bug description

Module: JTAG, Function: JTAG mailbox handshake mechanism synchronization loss

The JTAG mailbox handshake mechanism can lose synchronization if the CPU accesses the mailbox (read or write on JMB data registers) while the JMBOUTCTL/JMBIBCTL register is accessed via the JTAG interface.

Workaround:
1) Avoid reading or writing on JMB data registers via CPU while the JMBOUTCTL/JMBIBCTL register is accessed via the JTAG interface.
or
2) Send a request (set OUTREQ or INPREQ bit in JMBINCTL register) only when it can be served immediately. In detail, this means:

INPUT: Wait until the input registers are empty (IN0RDY = IN1RDY = 1) before sending an input request (INPREQ = 1) to the JMBINCTL register.
OUTPUT 16 Bit: Wait until the CPU has written register JMBOUT0 (OUT0RDY = 1) before sending an output request (OUTREQ = 1) to the JMBINCTL register.
OUTPUT 32 Bit: Wait until the CPU has written registers JMBOUT0 and JMBOUT1 (OUT0RDY = OUT1RDY = 1) before sending an output request (OUTREQ = 1) to the JMBINCTL register.

Detailed Bug Description (continued)

JTAGx JTAGx - Bug description

Module: JTAG, Function: JTAG 4-wire initialization is required to be run multiple times

To ensure entry into 4-wire mode after power up, repeat the 4-wire initialize multiple times.

Workaround:
Run initialization multiple times until initialization occurs

LPMx LPMx - Bug description

Module: LPM3 with ACLK = VLO

To achieve specified LPM3 current consumption using the VLO, the REFO oscillator is not automatically disabled, resulting in higher current.

Workaround:
Switch MCLK and SMLCK to VLO before entering LPM3. Upon returning from LPM3, switch back to normal operating clock for MCLK and SMCLK for the application.

PMM1 PMM1 - Bug description

Module: PMM, Function: Interrupt or POR events delayed when using DMA

If a DMA transfer is active, any interrupt or POR event triggered by SVSx or SVMx is delayed until end of DMA transfer. No event is lost, it is only delayed.

Workaround:
None. To minimize this effect, avoid long durations of DMA transfers.

Detailed Bug Description (continued)

PMM2 PMM2 - Bug description

Module: PMM, Function: Incorrect set event for SVSxIFGs if PMM has power mode change from low to high power.

Bug only occurs with following PMM configuration and conditions (also see *MSP430x5xx Family User's Guide* figures 4-5 and 4-6):
- SVSxMD=0: Set of SVSxIFG flag is disabled if PMM is in low-power mode (PMM is in low-power mode if device is in LPM2/3/4 or PMM is forced to low power with PMMCMDx = 10b in the PMMCTL1 register).
- During PMM low-power mode, the supervised voltage level is violated and is still violated if PMM power condition is switched back to high power.

Under these conditions, an incorrect set event for the SVSxIFG flag may occur at once when switching back the PMM to high-power mode.

Workaround:
Because the SVSx is not needed in PMM low-power mode (if SVSHMD = 0), disable it with SVSxE = 0 before entering any PMM low-power condition. Enable it again if PMM is switched back to high-power mode.

PMM3 PMM3 - Bug description

Module: PMM, Function: Bit PMMRSTLPM5IFG in register PMMIFG is not readable

Bit PMMRSTLPM5IFG in register PMMIFG is not readable. User software read value is always 0. However, bit functionality is as expected. Write access is working, and reset of this bit by reading of according SYSRSTIV vector word operates as described in the user's guide.

Workaround:
Check the status of PMMRSTLPM5IFG flag by reading the register SYSRSTIV vector word.

RTC1 RTC1 - Bug description

Module: RTC

Writing values to RTC registers may not latch properly, resulting in failure of register update by user software.

Workaround:
None

SYS1 SYS1 - Bug description

Module: SYS, Function: User NMI can disturb the high priority System NMI (priority inversion)

A lower priority User NMI can disturb a higher priority System NMI, leading to a priority inversion. An IRQ cannot disturb a higher priority NMI.

Workaround:
None

Detailed Bug Description (continued)

SYS2 SYS2 - Bug description

Module: SYS, Function: Protection weakness in BSL flash address range

If BSL protection is enabled, only the lowest 16 bytes of the secured BSL part should be usable as entry area (jump table). Instead, this entry window (not secured from readout by CPU) can be seen on other addresses.
Following ranges: 0x01600 to 0x0160F and 0x01700 to 0x0170F (for BSLSIZE==0), and so on for the rest of BSL sizes.

Readout is only possible when CPU, JTAG, and DMA security are still in place, and attempting to access it using them results in a security reset.

If, in addition, RAM space is assigned to BSL area and secured, the same mechanism is seen.
The following area should be protected : 0x01C00 to 0x01C0F.
In addition, following RAM areas are also secured and no longer accessible:
0x01D00 to 0x01D0F, 0x01E00 to 0x01E0F, 0x01F00 to 0x01F0F, 0x02000 to 0x0200F, ...
0x02B00 to 0x02B0F.
Any access leads to a security reset.

Workarounds:
- No use of RAM as secured area
- Do not put sensitive user code/data in the additional BSL entry address windows.

TB20 TB20 - Bug description

Module: Timer_B, Function: ACLK cannot be used as capture input

The capture/compare input of Timer_B7, CCI6B, according to the user guide is connected internally to ACLK. For the MSP430F543x series, CCI6B is connected only to TB6.

Workaround:
ACLK can be used indirectly by routing the ACLK signal externally via any port containing the ACLK output signal and feeding it back into TB6.

TBx TBx - Bug description

Module: Timer_B, Function: TBOUTH not functional

When P7.2 is configured as TBOUTH (P7.2SEL = 1 and P7.2DIR = 0), driving TBOUTH high does not switch the TB output pins to the high-impedance state as specified.

Workaround:
None

Detailed Bug Description (continued)

UCS1 UCS1 - Bug description

Module: UCS

When using VLO cock as WDT source clock, the WDT does not stop in a breakpoint condition. The VLO clock is not controllable in debug mode (no standard MSP430 module source clock), which results in unexpected resets or counter values from the WDT.

Workaround:
Do not use the VLO clock for WDT debug. In debug sessions for other modules, the WDT is switched off.

UCS2 UCS2 - Bug description

Module: UCS

When MCLK is slower than ACLK and an LPM mode is entered, it is possible that MCLK will remain in a high state, resulting in additional current being consumed.

This applies when the MCLK clock divider is greater than the ACLK cock divider setting, when using the same source. For example, if REFO is the source for both ACLK and MCLK, but MCLK is dividing by two, while ACLK is not.

Workaround:
None

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.

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 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