MSP430FG43x Device Erratasheet

Current Version

Devices	Rev:	CPU4	FLL3	OA1	TA12	TA16	TB2	TB16	US15	WDG2
MSP430FG437	D	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430FG438	D	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430FG439	D	✓	✓	✓	✓	✓	✓	✓	✓	✓
	C	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: See Appendix for prior revisions

Package Markings

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

CPU4 CPU4 - Bug description:

Module: CPU, Function: PUSH #4, PUSH #8

The single operand instruction PUSH cannot use the internal constants (CG) 4 and 8. The other internal constants (0, 1, 2, -1) can be used. The Assembler version 1.08 and higher produces correct code. The number of clock cycles is different:
PUSH #CG uses address mode 00, requiring 3 cycles, 1 word instruction
PUSH #4/#8 uses address mode 11, requiring 5 cycles, 2 word instruction

Workaround implemented in assembler.
No fix planned.

FLL3 FLL3 – Bug description:

Module: FLL+, Function: FLLDx = 11 for /8 may generate an unstable MCLK frequency

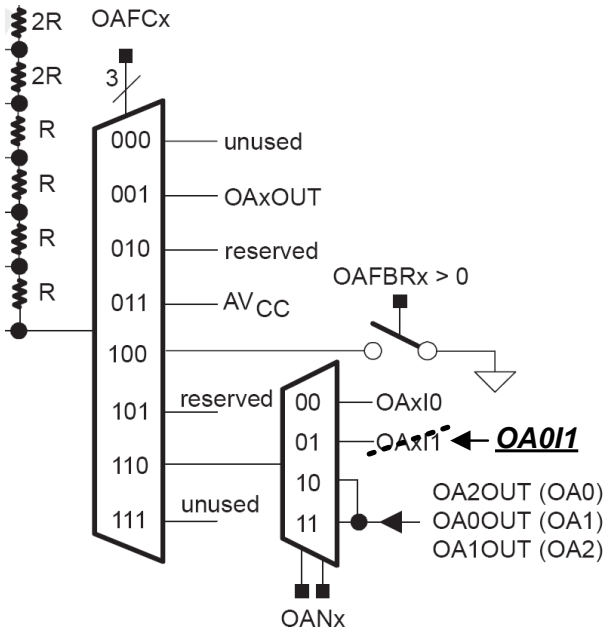
When setting the FLL to higher frequencies using FLLDx = 11 (/8) the output frequency of the FLL may have a larger frequency variation (e.g. averaged over 2sec) as well as a lower average output frequency than expected when compared to the other FLLDx bit settings.

Workaround:
None

OA1 OA1 - Bug description:

Module: OA, Function: OAxI1 input selection

Referring to the OA block diagram in the MSP430x4xx User's Guide, the internal connection of the OAxI1 input to the OAFCx mux is incorrect. The signal input to the OAFCx mux when OANx = 01 is OA0I1 for all OAs. See the figure below for a graphical representation:



Workaround:
None

Detailed Bug Description (continued)

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

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 afterwards.

Detailed Bug Description (continued)

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

US15 US15 - Bug description:

Module: USART0, USART1, Function: UART receive with two stop bits

USART hardware does not detect a missing second stop bit when SPB = 1.
The Framing Error Flag (FE) will not be set under this condition and erroneous data reception may occur.

Workaround:
None (Configure USART for a single stop bit, SPB = 0)

WDG2 WDG2 - Bug description:

If a key violation is caused by incorrectly accessing a flash control register, the watchdog interrupt flag is set in addition to a correctly generated PUC.

Workaround:
None

Appendix: Prior Versions

Devices	Rev:	CPU4	FLL3	OA1	SVS2	TA12	TA16	TB2	TB16	US15	WDG2
MSP430FG437	D	✓	✓	✓		✓	✓	✓	✓	✓	✓
	C	✓	✓	✓		✓	✓	✓	✓	✓	✓
	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430FG438	D	✓	✓	✓		✓	✓	✓	✓	✓	✓
	C	✓	✓	✓		✓	✓	✓	✓	✓	✓
	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
MSP430FG439	D	✓	✓	✓		✓	✓	✓	✓	✓	✓
	C	✓	✓	✓		✓	✓	✓	✓	✓	✓
	B	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Detailed Bug Description

SVS2 SVS2 - Bug description:

Module: SVS, DAC1: DAC1 overwrites an input of the SVS comparator

DAC1 overrides the input of the SVS comparator. This is caused by a conflict between SVS and DAC1 at Port 6.7. DAC1 is enabled when DAC12AMPx is > 0.

Workaround:
Do not enable DAC1 when SVS is used.