Flash Read Error and Susceptibility for MSP430F54xxA

MSP430

1 Overview

This document provides a description of the issue referred to as “Flash Read Error” impacting the current version of the MSP430F54xxA device derivatives. It also provides guidance in assessing whether and the extent to which the issue may affect a given application. Specific actions and customer recommendations are set forth in the PCN #20100910003A.

The flash read error results in the possibility that invalid data may be read from the Flash memory by the CPU. The issue is not related to Flash bit (cell) corruption: the Flash is valid as programmed. During Flash access, however, it is possible that data read out of the Flash by the CPU is corrupted. This may be undetected in the application with no ill effects; it could also result in an application error due to incorrect instruction execution or incorrect data fetch.

The root cause of the flash read error is understood and implementation of the fix within the Flash array has been completed. Device revisions incorporating this fix are currently in-process. For more information refer to the PCN.

The remainder of this document is intended to better describe the read error, how it can manifest within an application and recommendations for assessing the effects.

2 Affected Devices

The flash read error affects the MSP430F54xxA device derivatives. The table below lists all affected devices by orderable part number and can also be found in the “Product Affected” section of the PCN.

Table 1. Affected Devices

MSP430F5418AIPN	MSP430F5435AIPNR	MSP430F5438ACY	MSP430BT5190IPZ
MSP430F5418AIPNR	MSP430F5436AIPZ	MSP430F5438ACYS	MSP430BT5190IPZR
MSP430F5419AIPZ	MSP430F5436AIPZR	MSP430F5438AIPZ	MSP430BT5190IZQWR
MSP430F5419AIPZR	MSP430F5436AIZQW	MSP430F5438AIPZR	MSP430BT5190IZQWT
MSP430F5419AIZQW	MSP430F5436AIZQWR	MSP430F5438AIZQW	
MSP430F5419AIZQWR	MSP430F5436AIZQWT	MSP430F5438AIZQWR	
MSP430F5419AIZQWT	MSP430F5437AIPN	MSP430F5438AIZQWT	
MSP430F5435AIPN	MSP430F5437AIPNR	MSP430F5438AGACYS	

Note: For all listed part numbers, affected silicon revisions are up to and including Revision “D”

3 Description

The flash read error is a read out of Flash memory by the CPU of certain data bits as a logic “1” when the actual value stored in the flash memory bit location is a logic “0”.

3.1 Bus Architecture

To better describe the flash read error mechanism, it is important to understand the device architecture. The MSP430F54xxA implements a 32-bit cache for memory read operation – e.g. a physical read of the flash memory is performed on 32-bit boundaries; every 4 bytes, which is stored in a cache logic array. These 32-bit values are referred to as “double words”. Memory access by the CPU to individual bytes and words is then done by accessing the cache. This cache architecture is shown in Figure 1.

Note: The addresses used below are for example only and not intended to show specific susceptibility to the flash read error.

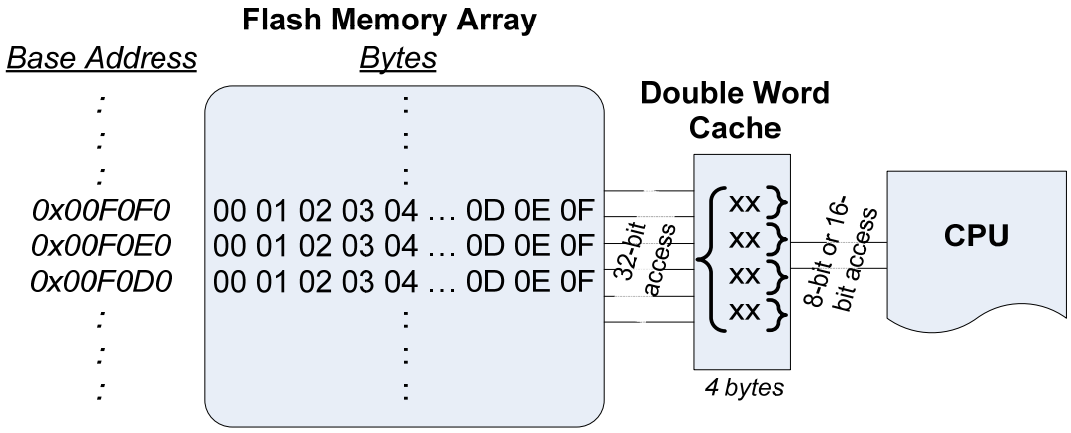

Figure 1. Simplified Cache Architecture and CPU Access

3.2 Affected Bits

The flash read error only affects the read access of the most significant bit (MSB) of each 32-bit Flash access by the cache. No other bit locations are affected. Each 31st bit location - i.e. the MSB of the 4th byte for each double word access, is susceptible to being read as “1” when it should be “0”. Bit locations for an entire 32-bit double word are defined as 0 to 31.

Furthermore, the issue only affects the 32-bit MSB when the flash bit is programmed as a “0”. In cases where the actual value stored is logic “1”, the bit will always read valid as logic “1” and is not susceptible to an invalid read.

Example: This illustration considers 16 bytes of flash data starting at address 0x0F0F0 and all contain the value 0x00. A CPU instruction execution triggers an access of the Flash word (at 0x0F0FE) and saves it to CPU register R5:

Instruction: `mov.w &0x0F0FEh, R5`

Result: Valid: R5= **0**000h
Invalid: R5= **8**000h

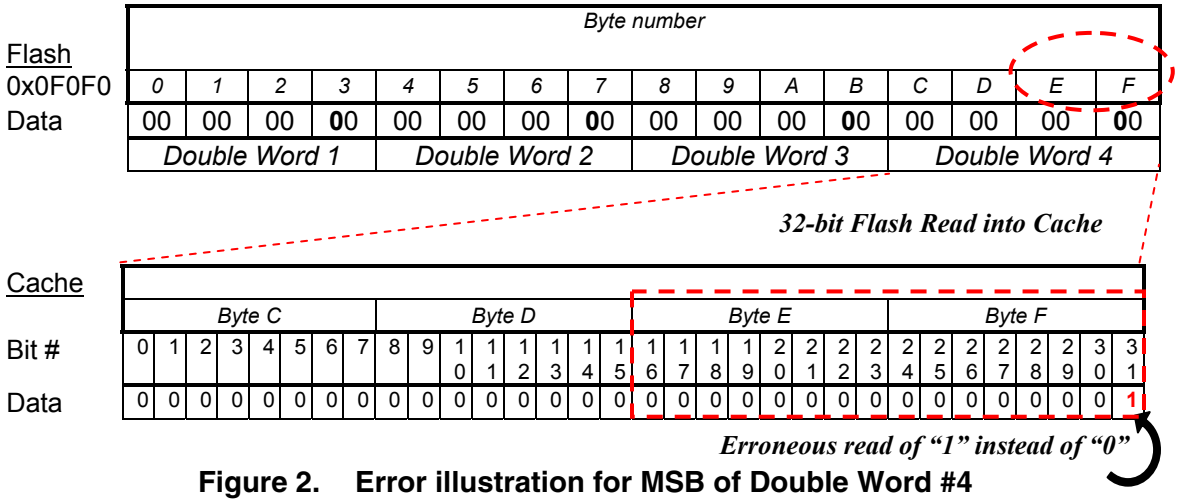

Figure 2. Error illustration for MSB of Double Word #4

The read operation by the CPU of address 0x0F0FE will actually trigger the reading of double word #4 from flash (data from 0x0F0FC to 0x0F0FF), which is then stored into the 32-bit cache. The uppermost two bytes (0x0F0FE and 0x0F0FF) make up the 16-bit word located at 0x0F0FE. The result of the read in the case of a flash read error is an invalid readout of the MSB for the given double word access. Instead of the expected logic "0", a logic "1" may be erroneously read out and stored in the cache for CPU computation: 0x8000 is erroneously read out and moved to R5 instead of 0x0000. This potential error applies to the MSB for each double word as indicated by the bold "**0**"s in Figure 2 at each 4th byte.

3.3 Affected Memory Locations

This flash read error is limited to specific memory locations for affected devices. MSP430F54xxA device derivative flash memory organization is based on 4 banks with 4 info and 4 BSL segments included. Each bank can be up to 64KB in size and are combined to create the different device memory configurations. Table 2 sets forth each device's memory configuration and affected banks.

Table 2. Affected Flash Memory Locations for MSP430F54xxA Device Derivatives

		MSP430F5419A ***MSP430F5418A***	***MSP430F5436A*** ***MSP430F5435A***	***MSP430F5438A*** ***MSP430F5437A*** ***MSP430BT5190***
Memory (flash)	*Total Size*	*128 KB*	*192 KB*	*256 KB*
Main: code memory	*Flash range*	*025BFFh-005C00h*	*035BFFh-005C00h*	*045BFFh-005C00h*
Main: code memory	Bank 3	N/A	23 KB 035BFFh-030000h	64 KB 03FFFFh-030000h
	Bank 2	**23 KB 025BFFh-020000h**	**64 KB 02FFFFh-020000h**	**64 KB 02FFFFh-020000h**
	Bank 1	64 KB 01FFFFh-010000h	64 KB 01FFFFh-010000h	64 KB 01FFFFh-010000h
	Bank 0	**41 KB 00FFFFh-005C00h**	**41 KB 00FFFFh-005C00h**	**64 KB 045BFFh-040000h 00FFFFh-005C00h**
Information memory (flash)	**Info A**	**128 B 0019FFh-001980h**	**128 B 0019FFh-001980h**	**128 B 0019FFh-001980h**
	Info B	128 B 00197Fh-001900h	128 B 00197Fh-001900h	128 B 00197Fh-001900h
	Info C	**128 B 0018FFh-001880h**	**128 B 0018FFh-001880h**	**128 B 0018FFh-001880h**
	Info D	128 B 00187Fh-001800h	128 B 00187Fh-001800h	128 B 00187Fh-001800h
Bootstrap loader (BSL) memory (flash)	BSL 3	512 B 0017FFh-001600h	512 B 0017FFh-001600h	512 B 0017FFh-001600h
	BSL 2	**512 B 0015FFh-001400h**	**512 B 0015FFh-001400h**	**512 B 0015FFh-001400h**
	BSL 1	512 B 0013FFh-001200h	512 B 0013FFh-001200h	512 B 0013FFh-001200h
	BSL 0	**512 B 0011FFh-001000h**	**512 B 0011FFh-001000h**	**512 B 0011FFh-001000h**

	affected
	not affected

Only bank 0, bank 2, info A, info C, BSL 0 and BSL 2 are susceptible to the flash read error as highlighted. Within these banks, the MSB locations for each double word can read out incorrectly when programmed as “0”. Flash program and data stored within bank1, bank 3, info B, info D, BSL 1 and BSL 3, independent of the value, will always be accessed correctly and are not susceptible to this issue.

3.4 Affected Flash Idle Time

The entire flash memory can be separated into two groups: affected and unaffected:

- Affected Group: bank 0/2, Info A/C, BSL 0/2 ***(affected locations)***
- Unaffected Group: bank 1/3, Info B/D, BSL 1/3 (unaffected locations)

Further narrowing the scope of this issue, this flash read error only affects the first flash access of the affected group that occurs after some period of time during which no flash accesses within the affected group has occurred. This period of no access to the affected flash is referred to as the “idle” time or t_IDLE. Additional double word reads from affected flash following the first read after the defined idle time are not subject to the flash read error and will be accessed correctly by the cache when the CPU clock is running at a frequency above 10kHz.

Any affected group address flash access results in all affected flash being active. Furthermore, all affected flash is in the idle state during access to any unaffected flash location.

Testing has shown that when the device is operating at 30°C the idle time, beyond which the read error may occur, is typically 15 msec. Any read of an affected bit after an idle time that exceeds 15 msec at 30°C is potentially susceptible to this issue. This idle time reduces to 0.5 msec typical at 85°C. When flash access has been inactive for longer than this idle time the issue can occur on the first double word read upon resuming access to the flash.

4 Application Affects

The way in which a given application may be affected if the flash read error occurs is dependent upon what the corrupted data accessed is intended to be used for. This section outlines the basic ways in which the read error can manifest into improper device functionality.

NOTE: Device power up IS NOT susceptible to the flash read error described in this document. The first flash access after power up to execute device bootcode is not affected.

Resetting the device via the RST pin can also be affected by the flash read error. When RST is asserted flash access will stop. When the assertion time of the RST signal is longer than the idle time discussed in section 3.4, the first flash access after RST is released can be affected. Flash will first be accessed by the CPU at the location containing the factory programmed bootcode. The bootcode is always executed after a BOR and loads factory stored calibration values and assesses the BSL configuration. If the flash read error occurs the access to start the bootcode is invalid and the device will enter a LPM4-like state without beginning user code execution, requiring reset or power cycle to restart.

After the device has properly powered up, the flash read error can occur during flash access through CPU instruction execution – e.g. active mode operation. This typically occurs only after flash access has been idle for a time greater than the idle time discussed in section 3.4. Operation usage for this to occur in-application can be illustrated into four use-cases: (i) entering and exiting LPMx modes from Flash, (ii) switching between active operation from RAM and Flash, (iii) using while(1), (iv) using the DMA, (v) erasing flash in-system, (vi) moving between groupings of flash during execution and (vii) LPM4.5 usage.

4.1 Low-Power Mode (LPMx) Use

A commonly used feature of the MSP430 that results in periods of flash access inactivity is the low power mode operation of the CPU. When using any of the LPMx modes built into the MSP430 architecture flash access is halted until the CPU is awakened via interrupt. Waking on any interrupt resumes active mode operation and consequently flash access for program execution.

When any low power mode is entered for a time that exceeds the idle time discussed in section 3.4, the first flash access made after that time can be affected by the read error. This will occur through the servicing of any enabled interrupt source. The first flash access made when an interrupt occurs is the access to fetch the 16-bit address pointer stored at the given ISR (interrupt service routine) vector location in flash. Table 3 shows the ISR vector location for each interrupt source in the affected device derivatives and highlights potentially affected ISR vector locations.

Table 3. MSP430F54xxA Device Derivative Interrupt Vector Addresses

Interrupt Source	Word Address	Lower Byte	Upper Byte	
System Reset *(e.g. Power-Up, RST, WDT Timeout)*	0FFFEh	0FFFEh	***0FFFFh***	*32-bit Double Word*
System NMI *(e.g. PMM, Vacant Memory Access)*	0FFFCh	0FFFCh	0FFFDh	
User NMI *(e.g. OSC Fault, Flash Access Violation)*	0FFFAh	0FFFAh	***0FFFBh***	*32-bit Double Word*
TB0	0FFF8h	0FFF8h	0FFF9h	
TB0	0FFF6h	0FFF6h	***0FFF7h***	*32-bit Double Word*
Watchdog Timer *(Interval Mode)*	0FFF4h	0FFF4h	0FFF5h	
USCI_A0 Receive/Transmit	0FFF2h	0FFF2h	***0FFF3h***	*32-bit Double Word*
USCI_B0 Receive/Transmit	0FFF0h	0FFF0h	0FFF1h	
ADC12_A	0FFEEh	0FFEEh	***0FFEFh***	*32-bit Double Word*
TA0	0FFECh	0FFECh	0FFEDh	
TA0	0FFEAh	0FFEAh	***0FFEBh***	*32-bit Double Word*
USCI_A2 Receive/Transmit	0FFE8h	0FFE8h	0FFE9h	
USCI_B2 Receive/Transmit	0FFE6h	0FFE6h	***0FFE7h***	*32-bit Double Word*
DMA	0FFE4h	0FFE4h	0FFE5h	
TA1	0FFE2h	0FFE2h	***0FFE3h***	*32-bit Double Word*
TA1	0FFE0h	0FFE0h	0FFE1h	
I/O Port P1	0FFDEh	0FFDEh	***0FFDFh***	*32-bit Double Word*
USCI_A1 Receive/Transmit	0FFDCh	0FFDCh	0FFDDh	
USCI_B1 Receive/Transmit	0FFDAh	0FFDAh	***0FFDBh***	*32-bit Double Word*
USCI_A3 Receive/Transmit	0FFD8h	0FFD8h	0FFD9h	
USCI_B3 Receive/Transmit	0FFD6h	0FFD6h	***0FFD7h***	*32-bit Double Word*
I/O Port P2	0FFD4h	0FFD4h	0FFD5h	
RTC_A	0FFD2h	0FFD2h	***0FFD3h***	*32-bit Double Word*
Reserved	0FFD0h	0FFD0h	0FFD1h	
	0FFxxh	*– Susceptible address*		

Interrupt vector addresses highlighted in Table 3 are susceptible to the flash read error. When the CPU wakes to service a susceptible ISR after the idle time discussed in section 3.4 has been exceeded in any LPMx mode, the MSB of the upper affected byte may be corrupted as a “1”. If this occurs the CPU will vector to a location that is not the valid start address for the given ISR and potentially execute code erroneously.

4.2 CPU Code Execution from RAM and Flash

In applications where the CPU is redirected in user code to execute from RAM instead of flash, the read error can occur upon resumption of flash access; either program or data access within the flash memory range. Code executed from RAM that does not make any flash accesses for a time exceeding the idle time discussed in section 3.4 can be subject to the flash read error once flash access resumes, causing an invalid data or instruction fetch to occur. In cases where interrupts are also active while code is executed from RAM, the same scenarios may apply as outlined in section 4.1.

4.3 while(1) Use (Also JMP$ in Assembly)

Use of while(1) in C (which corresponds to the assembly equivalent `JMP$`; e.g. “jump to self”) can result in flash idle times even though the CPU is active. This is due to the manner in which the CPU and cache operate and the placement of the JMP$ instruction within flash.

Such operation may be affected by the flash read error depending on memory position of the `JMP$` instruction. The read error is possible when the `JMP$` instruction resides in the lower word of a double word fetch from flash allowing execution to happen completely from cache turning off flash access. In this case, when the execution of the `JMP$` exceeds the idle time discussed in section 3.4 prior to servicing an interrupt, the device is susceptible to the flash read error with the next ISR service and the same scenarios outline in section 4.1 may apply.

Figure 3 shows the relationship of the `JMP$` instruction position in memory and the behavior of the Flash access. On the right, the `JMP$` is at the lower word of the double word fetched from flash. This placement results in the CPU being able to execute entirely from the cached double word containing the `JMP$` instruction which stops flash access activity. Execution of the `JMP$` in such a memory alignment for a time exceeding the idle time discussed in section 3.4 can result in a read error on the next flash access – e.g. servicing an active ISR.

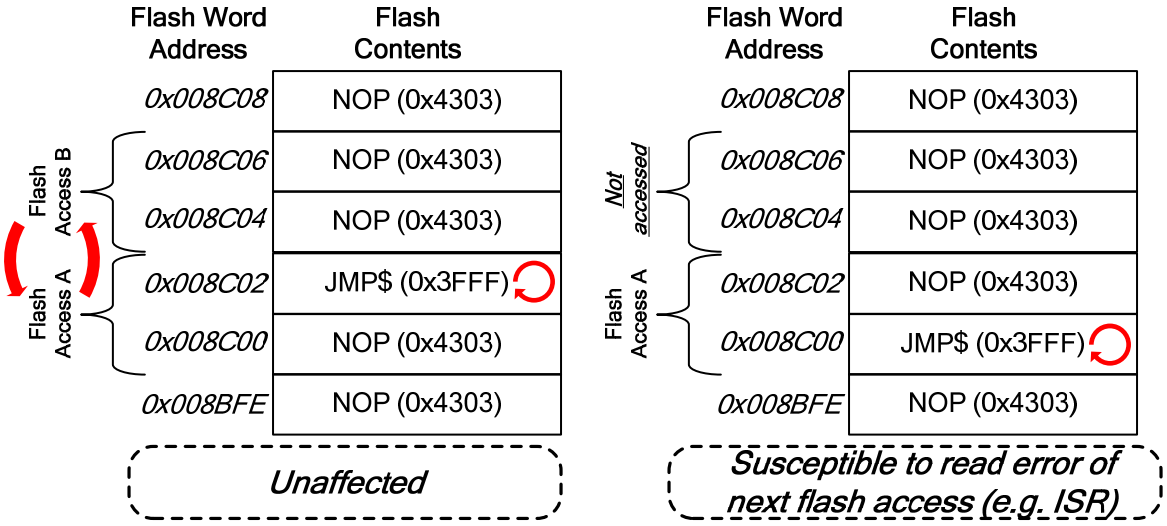

Figure 3. `JMP$` Alignment and Read Error Behavior

However, when the `JMP$` instruction resides in the upper word of the double word memory fetch, as shown in the left portion of Figure 3, the cache continuously alternates between flash accesses A and B which keeps the flash access constant and unaffected by the read error on a subsequent ISR service fetch.

4.4 DMA Use

The DMA can be used to read data from flash without any CPU activity and accesses the cache in the same way to make Flash accesses. DMA access can be configured to read flash and is triggered independently of any interrupt activity (e.g. Timers, ADC, USCI). When the DMA is configured to read from flash upon receiving a given trigger, the first read of flash may be affected by the read error when the idle time discussed in section 3.4 is exceeded.

4.5 In-System Flash Segment Erase

Completion of a flash segment/bank erase (e.g. via user code) is specified to take 23 msec minimum to 32 msec maximum in the device datasheet. During this time the flash is in the idle condition. This time exceeds the idle time discussed in section 3.4 and can result in a flash read error upon the next flash access executed after the erase command.

NOTE: Writing to flash results in a maximum idle time of 85usec, well below the time discussed in section 3.4 and is not affected.

4.6 Execution Between Banks, Info and BSL Groupings

Because flash active vs idle states are dependent upon the address being accessed, it is possible that affected flash locations are in idle while other areas of flash are being accessed. This address grouping dependency is outlined in section 3.4.

The entire flash memory can be separated into two groups: affected and unaffected. When one group is being accessed the other group is idle. CPU execution moving from unaffected flash into affected flash may be susceptible to a flash read error and is limited to the first access of the affected flash. This would be possible if the idle time as discussed in section 3.4 is exceeded during execution from the unaffected flash.

4.7 LPM4.5 Use

When the device is in LPM4.5, the flash is idle. When wake-up is triggered via the RST or GPIO pin, the device may be susceptible to the flash read error as in the case of a normal reset as described earlier.

5 Influencing Factors

There are numerous factors that influence the likelihood of the flash read error occurring during application operation. One factor is the operating temperature of the device and idle time of the flash as discussed in section 3.4. Table 4 summarizes the typical idle time durations beyond which the flash read error becomes more likely to occur.

Table 4. Flash Idle Time (t_IDLE) Variation With Temperature

Temperature [°C]	Idle Time Typical [ms]
30	15
50	2
85	0.5

Note : The relationship between temperature and idle time is exponential.

External supply voltage (DVcc, AVcc) and internal core voltage (Vcore) also have an influence on likelihood of whether the flash read error may occur. Figure 4 shows Vcc vs System (CPU) clock vs Vcore valid operating conditions per the device datasheet.

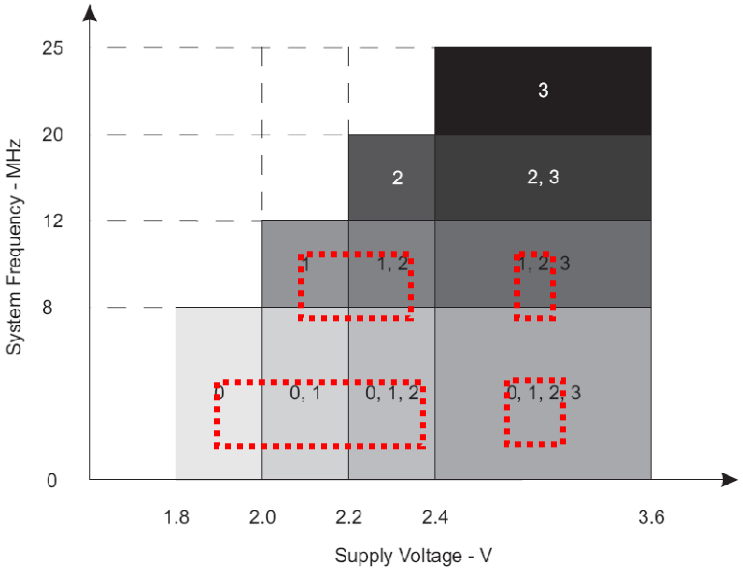

Figure 4. Core Voltage Settings With Reduced Susceptibility

The Vcore settings 0 and 1 and corresponding operating ranges are highlighted. These Vcore level settings result in reduced susceptibility to the flash read issue vs levels 2 and 3.

While all Vcore settings when used under the proper Vcc and clock frequency ranges are valid for device operation, usage of Vcore level 2 and 3 settings are twice as likely to result in flash read errors at susceptible bit locations compared to levels 0 and 1.

An additional influence on the likelihood of the flash read error occurring is process variation during fabrication of the devices. These variations are not detectable directly by the user but can result in variation of flash read error failure rates seen in production and the field.

6 Application Analysis

The following checklist highlights specific areas to assess within a given application to determine if the conditions previously outlined are present. It is not intended to be a complete list and sound judgment should be used in analyzing any specific application. All items should be considered independently.

☐ Is the longest time the CPU spends in any LPM mode (CPUOFF), executing code from RAM or from unaffected flash memory > the applicable t_IDLE at the expected operating temperature of the device (see Table 4 and section 3.4)?

- If no, the read error typically will not affect runtime operation

☐ Is while(1)/JMP$ being used?

- If yes, does the disassembled JMP$ (jump to self) instruction reside in the lower word of a 32-bit aligned double word in Flash?
  - If yes, is the time spent executing the while(1) > the applicable t_IDLE at the expected operating temperature of the device?
  - If no applies to any of the above questions, the read error typically will not affect runtime operation

☐ Is the DMA used to access affected flash source addresses as defined in Table 2?

- If no, the DMA susceptibility does not apply

☐ Are the Flash memory banks affected (see Table 2) by read error used in the application (for program or data)?

- If no, the read error will not affect runtime operation

Note: The interrupt vector address fetch locations reside in BANK0 which is affected. See the next checklist item.

☐ Are susceptible ISR vector address locations highlighted in Table 3 serviced after a time > the applicable t_IDLE at the expected operating temperature of the device?

- If no, the read error typically will not affect runtime operation

- ☐ If used do the 16-bit values at the ISR vector address locations highlighted as susceptible in Table 3 contain a “0” in the MSB?
    - If no, the read error will not affect valid interrupt vector fetching of these ISRs
    - For applications that exceed the applicable t_IDLE idle time at the expected operating temperature of the device while in LPM modes, RAM execution or while(1) from an affected address AND wake to interrupts:
        - The interrupts that exit the given mode if at the affected addresses in Table 3 and containing an MSB = 1, will not be affected by the flash read error.

7 Application Robustness

While it is not possible to completely eliminate the possibility of flash read error occurrence under all application use cases, there are steps that one can take to reduce susceptibility in instances in which changes to application functionality are acceptable.

7.1 Vcore

Operating the device at an internal core voltage level 0 or 1 can reduce the likelihood of flash read error occurrence by as much as 50% versus operation at level 2 and level 3.

7.2 Idle Time

Reducing idle time between flash accesses as low as possible will reduce the likelihood of flash read error occurring. Temperature range of the application should be taken into account given the variation on idle time relation to temperature.

7.3 ISR Placement

Manual placement of interrupt service routines into memory locations above 0x008000 can eliminate the effect on interrupt vector address fetches. When ISR locations cannot be manipulated, limiting interrupt wakeup to peripheral interrupt vectors that are fetched from addresses unaffected by the flash read error can also be leveraged to wake from extended idle times between flash accesses properly.

7.4 Program and Data Placement

Manual placement of all user code and data into memory locations within banks 1 and/or 3 eliminate the possibility that the flash read error affects this code.

7.5 Device Reset

To minimize the potential of encountering the flash read error effects upon reset of the device, issuance of 2 external reset pulses externally to the RST pin where the duration of each low reset pulse and the time between pulses is less than the idle time discussed in section 3.4 is an effective methodology. For worst case, a time of 0.5msec or less as shown in Figure 5 is recommended.

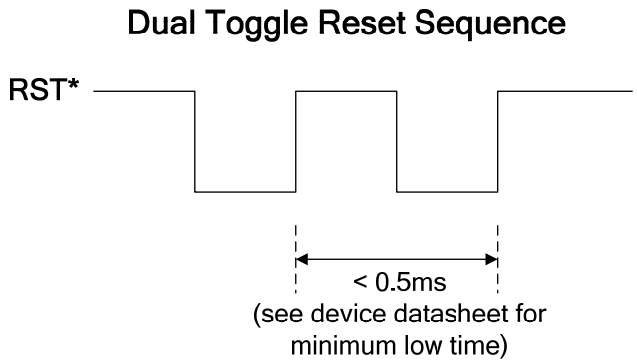

Figure 5. Recommended RST signal

7.6 In-System Flash Erase

To avoid flash read error susceptibility after an in-system Flash erase is completed, dummy instructions that include the MSB = “1” placed immediately after the flash erase instruction will result in proper flash execution. Three instances of the dummy instruction are required to account for the possible upper or lower double word alignment in flash. A short code example demonstrating this is provided.

```
#include "msp430f5438a.h"

void main(void)
{
  WDTCTL = WDTPW + WDTHOLD;                 // Stop WDT

  char *Flash_ptr = (char *)0x1800;         // Initialize Flash segment D ptr

  // Prepare Flash for erase operation
  FCTL3 = FWKEY;                            // Clear Lock bit
  FCTL1 = FWKEY + ERASE;                    // Set Erase bit

  // Erase Flash segment at Flash_ptr
  *Flash_ptr = 0;                           // Dummy write to erase Flash seg

  // Directly after the Flash erasure, in-line two assembly op-codes which have
  // the MSB of their 16-bit op-codes set. This will ensure that in case of a
  // 32-bit Flash read the MSB is already one (and hence will be unaffected by
  // any potential flash read error susceptibility). Three op-codes are needed
  // to cover both possible cases of alignment between the op-codes and the
  // 32-bit Flash fetch.
  // Note that operations with R3 as a destination register have no effect
  // (same as an actual NOP). Also, the 'bis.w' instruction was chosen since
  // it does not affect the status register (SR) bits neither.
  // It is recommended to verify the placement of these instructions right next
  // to the Flash erasure instruction using the debugger disassembly window.
  asm(" bis.w #0,R3 ");                     // Flash contents: 0x03 0xd3
  asm(" bis.w #0,R3 ");                     // Flash contents: 0x03 0xd3
  asm(" bis.w #0,R3 ");                     // Flash contents: 0x03 0xd3

  //  Lock the Flash again
  FCTL3 = FWKEY + LOCK;                     // Set LOCK bit

  while (1) {
      __no_operation();                     // Loop forever, SET BREAKPOINT HERE
  }
}
```

The three in-line assembly instructions above are the dummy instructions that will execute just after flash erase completion. These instructions have no impact to the CPU or functionality. They simply execute with no output and represent accesses to flash where the MSB is set to 1 within the instruction op-code. This assures the instructions execute properly even if a flash read error occurs.

7.7 BSL Entry and Exit

When entering and exiting the BSL, it may be possible that the device is susceptible to the flash read error. The effect is the same as is described for the reset pulse in section 4. In order to minimize the possibility of this occurrence following the entry and exit timings as shown in figures 6 and 7 are recommended.

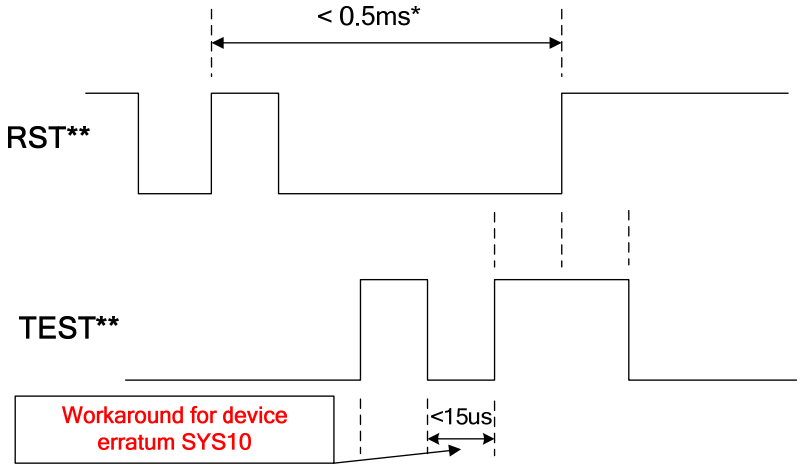

Figure 6. BSL Entry Sequence Recommended Timing

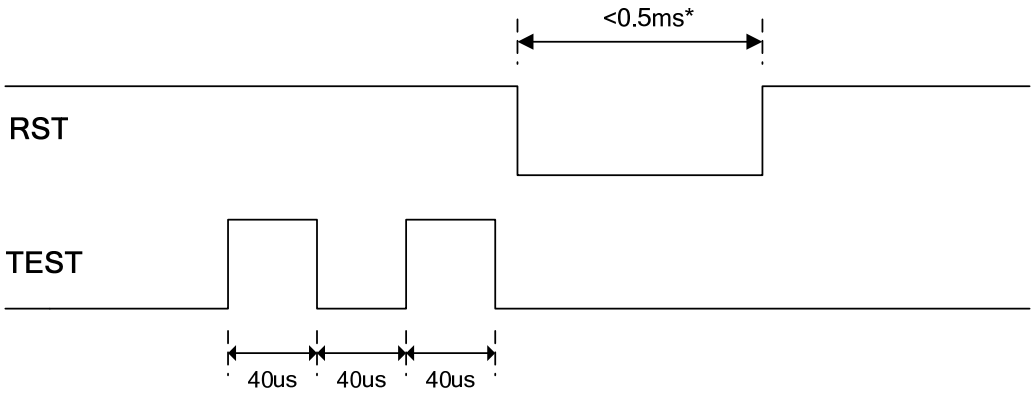

Figure 7. BSL Exit Sequence Recommended Timing

*Note: See device datasheet for minimum specified reset signal low time.

7.8 Application Testing

Production testing at the maximum specified temperature for the given application will provide the worst-case corner for occurrence of the flash read error. In addition, subjecting the application to expected operating conditions resulting in worst-case flash idle time duration will also enhance the likelihood of flash read error occurrence. While this exercise will not assure all potentially affected devices will be detected it is the most effective means to increase probability of detection.

8 Summary

The information in this document provides the reader with information that can be applied to assessing the susceptibility of an application to the flash read error described herein. In addition to the information presented here, the performance of the application being assessed should also be considered as an important data point in determining risk. Occurrence or not of failures in the field or at production of a given end equipment serve as an additional indicator of robustness and the likelihood of an application to flash read susceptibility.

9 Revision History

Section	Page	Description
Table 2	4	Added INFO and BSL memory sections
Section 3.4	5	Clarified flash access location relationship and idle time definition
Section 4	5	Removed power up from affected use-cases, added application use-cases
Table 4	10	Added note, “25” changed to “30”, 50C data point added
Section 7	12	Updated information for improving application robustness

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
DLP® Products	www.dlp.com	Communications and Telecom	www.ti.com/communications
DSP	dsp.ti.com	Computers and Peripherals	www.ti.com/computers
Clocks and Timers	www.ti.com/clocks	Consumer Electronics	www.ti.com/consumer-apps
Interface	interface.ti.com	Energy	www.ti.com/energy
Logic	logic.ti.com	Industrial	www.ti.com/industrial
Power Mgmt	power.ti.com	Medical	www.ti.com/medical
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Space, Avionics & Defense	www.ti.com/space-avionics-defense
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video and Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless-apps

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