Freescale Semiconductor
Device Errata

MSE5125_0M01S
Rev. 3, 12/2011

MPC5125 (M01S) Errata

by: Microcontroller Division

This document identifies implementation differences between the MPC5125, M01S (Rev. 1), and the processor's description contained in the *MPC5125 Microcontroller Reference Manual* and the *MPC5125 Microcontroller Data Sheet*. Refer to http://www.freescale.com for the latest updates.

Contents

1 Overview

Table 1. MPC5125 Errata Summary

Module	ID	Date Added	Title
e300 Core	467	12/9/2008	Section 2.1, e300 — DTLB (Data Translation Lookaside Buffer) LRU performance issue
	ERR 003383	12/01/2011	Section 2.2, If critical interrupt and normal interrupt are being used in a system, the e300 core may hang
PMC	496	03/18/2009	Section 3.1, Instability when entering deep sleep mode
RTC	471	02/02/2009	Section 4.1, The Tamper (TAMP) bit in the SRTC is not reliable
	509	01/21/2010	Section 4.2, RTC Oscillator may not start properly
TEMPSENSE	393	02/11/2009	Section 5.1, Temperature sensor does not provide specified accuracy
USB	42	08/14/2007	Section 6.1, Read of PERIODICLISTBASE after successive writes may return wrong data
	465	03/18/2009	Section 6.2, The USB controller can issue transactions that lock up the AHB bus under certain conditions
	476	03/18/2009	Section 6.3, The PCI bit of the USBSTS register is not triggered by port enable/disable change
	477	02/26/2009	Section 6.4, In Host mode, an RX FIFO overflow can generate a false babble error
	478	02/26/2009	Section 6.5, No interrupt generated on error condition in ISO Mult 3 mode
	479	02/26/2009	Section 6.6, Device ISO IN transactions underrun on ISO Mult 3 transfers
	480	03/17/2010	Section 6.7, Port change detect bit is set when a resume is forced by SW write to FPR bit in PORTSC register
	481	02/26/2009	Section 6.8, Incorrect value read from TXPBURST/RXPBURST
	485	03/18/2009	Section 6.9, Wrong DEVICEADDR register access in Host Mode
	484	02/26/2009	Section 6.10, Active bit not reset in ISO Mult 2 when packet size = total bytes
	486	03/26/2009	Section 6.11, Core Device fails when two OUT transactions occur in a short time
	501	03/17/2010	Section 6.12, OTG Controller as Host does not support Data-line Pulsing SRP
	503	03/17/2010	Section 6.13, Device does not respond to INs after receiving corrupted handshake from previous IN transaction
	504	03/17/2010	Section 6.14, Last read of the current dTD done after USB interrupt

2 e300 Core

2.1 e300 — DTLB (Data Translation Lookaside Buffer) LRU performance issue

2.1.1 Description

A performance degradation may occur for the e300 core under some specific load and store memory copy operations. This issue is related to performance only and the core always produces the correct and expected result. The issue relates to LRU during a DTLB miss with a load and a store operation within a 128 KB range (the EA[15:19] DTLB index) relationship between the source and the destination of a memory to memory copy. If the load *and* the store operations (going to a different address) are such that the EA[15:19] remain unchanged, then the condition exists, and the DTLB LRU remains pointing to a single way.

The effect of the issue is that the DTLB only uses a single way entry for all operations. This results in the Data TLB DSI exception being taken for *every* load and store within the range, not just the exception needed to load the initial page.

2.1.2 Workaround

The software workaround is to implement a Least Recently Written (LRW) in software, since the Least Recently Used (LRU) answer from the hardware will always point to way 1 after the first entry. Currently, when a Data Translation Miss exception occurs, the SRR1 bit 14 (DTLB replacement way) is automatically set to indicate the way selected for replacement. However, this can be overwritten by altering that bit with an LRW setting in software. The DTLB implementation is a 2-way set associative implementation with 32 entries per way. This means one word (32 bits) can be kept in software to manage the LRW settings on every new load to the DTLB. The LRU hardware algorithm is changed to an LRW software algorithm.

2.2 If critical interrupt and normal interrupt are being used in a system, the e300 core may hang

2.2.1 Description

If critical interrupt and normal interrupt are being used in a system, the e300 core may hang.

2.2.2 Workaround

Instead of using an RFI at the end of every exception handler, replace the RFI with the following:

Disable critical interrupts by setting MSR[CE] to 0 with **mtspr**.

Copy SRR0 and SRR1 to CSRR0 and CSRR1, respectively.

Execute an RFCI. This enables MSR[CE] and any other bits that original RFI would have set, including MSR[EE]

Sample code:

```
// Disable MSR[CE]
mfmsr   r2
lis     r3, 0xffff
ori     r3, r3, 0xff7f
  and     r2, r2, r3
  sync
  mtmsr   r2
  isync

// Copy SRR0, SRR1 to CSRR0 and CSRR1
  mfspr   r2, srr0
  mfspr   r3, srr1
  mtspr   csrr0, r2
  mtspr   csrr1, r3

… restore GPRs

rfci
```

3 PMC

3.1 Instability when entering deep sleep mode

3.1.1 Description

An instability with an internal counter in the PMC module may occasionally prevent the PMC from properly entering deep sleep mode.

3.1.2 Workaround

Do not use deep sleep mode.

4 RTC

4.1 The Tamper (TAMP) bit in the SRTC is not reliable

4.1.1 Description

The TAMP bit in the SRTC does not reliably provide an indication that the RTC power supply or the RTC oscillator has been interrupted.

4.1.2 Workaround

In your system’s design, do not rely on the tamper indication to be reliable.

4.2 RTC Oscillator may not start properly

4.2.1 Description

The Real-Time Clock (RTC) oscillator may not start properly when the V_{BAT_RTC} power domain is active and V_{DD_CORE} is transitioning from power-off to power-on.

When the V_{BAT_RTC} power domain is active and the V_{DD_CORE} transitions from off to on, the RTC time base becomes unreliable. The functionality of the oscillator and clock feeding the RTC counter is in one of three states:

- Normal operation
- Oscillator circuit disabled
  - The crystal-based circuit is not operational, preventing a clock to the RTC time base so the RTC cannot continue counting.
- The clock signal into the RTC is running at 32 kHz instead of 1 Hz.

If the RTC clock signal is not operational, it affects the following chip features:

- The RTC will not keep accurate time.
- The chip cannot enter or exit Hibernate mode. All Hibernate exit conditions are affected:
  - CAN activity
  - External GPIOs: GPIO00, GPIO01, GPIO02, and GPIO03
  - RTC alarm
- The chip cannot enter Deep Sleep mode.

4.2.2 Workaround

Use combinations of the following measures to obtain different levels of normal operation:

1. Ensure proper RTC operation.

   Ensure that the V_{BAT_RTC} power domain does not lead the V_{DD_CORE} by more than 3 ms, or toggle V_{BAT_RTC} after V_{DD_CORE} has been turned on. The 3 ms is measured from V_{BAT_RTC} at 50% to V_{DD_CORE} at 90% of full values.

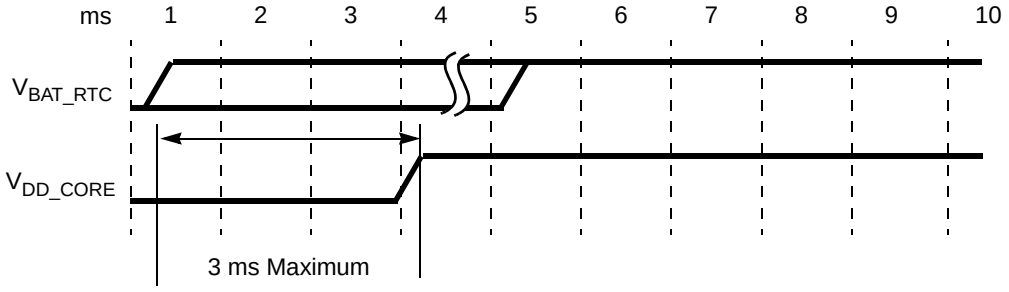

Figure 1. V_{BAT_RTC} to V_{DD_CORE} Timing

V_{BAT_RTC} can be powered concurrently with or enabled by V_{DD_CORE}. Cycling V_{BAT_RTC} power will result in the RTC counter being reset. See item 4 below. Following this workaround measure eliminates the need for measures 2 and 3 below.

2. Ensure the RTC time base is running.

   To ensure that the RTC input clock is operational, the RTC_XTALI input must be driven by an oscillator signal rather than using a crystal. For oscillator signal specifications, please see the *MPC5125 Datasheet (MPC5125)* specification number D3.20. A running RTC time base affects the ability to enter and exit Hibernation, and the ability to enter Deep Sleep power modes.

3. Ensure correct RTC time base.

   If V_{BAT_RTC} is kept active when V_{DD_CORE} is inactive, on each startup (e.g.,cold boot or resume from Hibernate), software must ensure that bits in the extended clock register (MBAR + 0x082C) are correct. To ensure the value is correct, software must write the following values, in sequence, into the extended clock register at address MBAR + 0x82C:

   a) 0x8000
   b) 0xC000
   c) 0x8000
   d) 0x0000

   This sequence sets the extended register into normal operational mode, which will operate at the proper frequency (32 kHz). This accelerated time base will affect proper wakeup times from Hibernation and Deep Sleep power modes.

   NOTE

   The Extended Register at MBAR + 0x82C is currently not documented in the *MPC5125 Reference Manual (MPC5125RM)*.

4. Restore an accurate real time.

   If the real time is lost due to the RTC counter being reset which is required by the application, the system needs to use an alternate real time clock reference such as an RTC clock chip, or retrieve the time from an external source such as the internet.

5 TEMPSENSE

5.1 Temperature sensor does not provide specified accuracy

5.1.1 Description

The temperature measurement accuracy is ±10 °C compared to the specification target of ±5 °C.

5.1.2 Workaround

When utilizing the temperature sensor, be sure to account for the accuracy in the system.

6 USB

6.1 Read of PERIODICLISTBASE after successive writes may return wrong data

6.1.1 Description

The USB controller allows a preset device address in DEVICEADDR register before the device is enumerated, using a shadow register, to assist slow processors. The problem is that this mechanism, which is supposed to be functional only in device mode, is not blocked in host mode. The DEVICEADDR register serves as PERIODICLISTBASE in host mode.

If PERIODICLISTBASE was set to some value, and later it is modified by software in such a way that bit 24 is set to a 1, then an incorrect (previous) value is read back.

6.1.2 Workaround

Write 0x0000 to the PERIODICLISTBASE two times in succession before setting PERIODICLISTBASE to the desired value. This clears the shadow register.

6.2 The USB controller can issue transactions that lock up the AHB bus under certain conditions

6.2.1 Description

The AHB bus can lock up if there are other pending transactions when the USB controller issues a specific bus transaction. These specific transactions can happen when the packet size modulo BURSTSIZE setting is 5, 6, or 7 bytes.

6.2.2 Workaround

In the system software, set the USB SBUSCFG register to INCR4/INCR8/INCR16 (0x01/0x02/0x03) to avoid any problematic transfers.

6.3 The PCI bit of the USBSTS register is not triggered by port enable/disable change

6.3.1 Description

The PCI bit of the USBSTS register is not set to 1 when a port is enabled or disabled by a set/clear of the PP or PE bit in the PORTSC register.

6.3.2 Workaround

The software cannot rely on the PCI interrupt to notify it if the port was enabled/disabled by set/clear of the PP or the PE bit in the PORTSC register.

6.4 In Host mode, an RX FIFO overflow can generate a false babble error

6.4.1 Description

When the USB interface is configured as a host (or A-device in OTG), an RX FIFO overflow can falsely generate a frame babble error. The frame babble error causes the offending port to be disabled.

6.4.2 Workaround

In such case, to enable the port again the software stack must reset the bus and re-enumerate the connected device.

6.5 No interrupt generated on error condition in ISO Mult 3 mode

6.5.1 Description

When an ISO Fulfillment Error occurs and Mult = 3, the status is not updated to indicate such error and an interrupt is not fired. If the USB interface is functioning in Device ISO IN mode, it does not force the retirement of the ISO-dTD, so it transmits such data in the next uFrame. All the transactions following this situation will be shifted by one in the Mult order until all dTDs are consumed.

6.5.2 Workaround

Use Mult = 2.

6.6 Device ISO IN transactions underrun on ISO Mult 3 transfers

6.6.1 Description

Device ISO IN transactions might underrun on high-speed high-bandwidth endpoints that transfer more than 2048 bytes per period (Mult = 3).

6.6.2 Workaround

Limit the transfer to 2048 bytes (Mult = 2).

6.7 Port change detect bit is set when a resume is forced by SW write to FPR bit in PORTSC register

6.7.1 Description

Working as host, when doing resume (software is driven by writing to bit 6 of PORTSC*x* register), a port change interrupt was generated at the end of resume. According to the EHCI specification, no interrupt should be generated.

6.7.2 Workaround

The extra interrupt should be taken into account when doing resume.

6.8 Incorrect value read from TXPBURST/RXPBURST

6.8.1 Description

The value of the USB register BURSTSIZE (base + 0x160) is incorrectly read when the field AHBRST of USB register SBUSCFG (base + 0x090) is set to any value between 0b000 and 0b100.

6.8.2 Workaround

Create a software register that shadows the value of BURSTSIZE. This shadow register should be used for reading back values.

6.9 Wrong DEVICEADDR register access in Host Mode

6.9.1 Description

In Host mode, when reading register PERIODICLISTSBASE (Base + 0x0154), the read value corresponds to the DEVICEADDR instead of the PERIODICLISTBASE.

6.9.2 Workaround

Create a software register that shadows the value of PERIODICLISTBASE. Use this shadow register for reading back values.

6.10 Active bit not reset in ISO Mult 2 when packet size = total bytes

6.10.1 Description

In Device mode, for ISO IN transfers when the combination of the Mult value and the Max Packet Size results in a larger number of packets than what is set in Total Bytes, then the Active bit will not go to 0 when the transfer is done.

6.10.2 Workaround

The software should make sure that the Mult value correctly reflects the total number of packets.

6.11 Core Device fails when two OUT transactions occur in a short time

6.11.1 Description

When the controller is configured as Device and is receiving two consecutive OUT transactions, such as the sequence OUT – DATA0 – OUT – DATA1, if the interpacket delay between DATA0 and the second OUT is less than 200 ns, the Device will see the DATA1 packet as a short-packet even if it is correctly formed. This terminates the transfer from the Device point of view, generating an IOC interrupt.

6.11.2 Workaround

In such a situation, software must check whether the transferred amount of bytes is correct or not, and then proceed accordingly.

6.12 OTG Controller as Host does not support Data-line Pulsing SRP

6.12.1 Description

When the OTG core is acting as a Host and VBUS is turned off, and the attached Device attempts to perform a Session Request Protocol by using Data-line Pulsing, it will not be recognized by the Host. Also, when doing HNP and becoming a Host, a SE0 is forced in the line causing the OPT TD5.4 test to fail, without software workaround.

6.12.2 Workaround

A software workaround is possible for the HNP situation only. With this workaround, it is possible to pass OPT TD5.4 test. The software must assert core mode to the device (USBMODE.CM) and set the Run/Stop bit (USBCMD.RS) to 1 just after the controller ends reset. Wait until USBCMD.RST is 0 after setting it to 1.

6.13 Device does not respond to INs after receiving corrupted handshake from previous IN transaction

6.13.1 Description

The device does not respond to IN tokens after receiving a corrupted handshake if the next tokens arrive sooner than the BTO constant as defined on vusb_hs_pkg (BTO_TIME_HS_60MHZ / 30MHZ).

6.13.2 Workaround

No workaround is available.

6.14 Last read of the current dTD done after USB interrupt

6.14.1 Description

After executing a dTD, the device controller executes a final read of the dTD terminate bit to see if another dTD has been added to the linked list by software right at the last moment.

It was found that the last read of the current dTD is being done after the interrupt is issued and not before. There could be a race between this final dTD read and the interrupt handling routine servicing the interrupt on complete that may result on the software freeing the data structure memory location, prior to the last dTD read being performed.

6.14.2 Workaround

No workaround is available.

Table 2. Revision History Table

Rev. Number	Substantive Changes	Date of Release
1.0	• Initial release.	03/2009
2.0	• Added item 509, RTC. • Added items 480, 501, 503, 504, USB. • Removed item 469, PSC.	03/2010
3.0	• Added item ERR003383, e300 Core.	12/2011

How to Reach Us:

Home Page:
www.freescale.com

Web Support:
http://www.freescale.com/support

USA/Europe or Locations Not Listed:
Freescale Semiconductor, Inc.
Technical Information Center, EL516
2100 East Elliot Road
Tempe, Arizona 85284
1-800-521-6274 or +1-480-768-2130
www.freescale.com/support

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
www.freescale.com/support

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064
Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor China Ltd.
Exchange Building 23F
No. 118 Jianguo Road
Chaoyang District
Beijing 100022
China
+86 10 5879 8000
support.asia@freescale.com

Freescale Semiconductor Literature Distribution Center
1-800-441-2447 or +1-303-675-2140
Fax: +1-303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Document Number: MSE5125_0M01S
Rev. 3
12/2011

freescale™ semiconductor