MAC7106MSE1
Rev. 1, 04/2005

MAC71*x*6 Microcontroller Device Mask Set Errata

This document identifies implementation differences (summarized in Table 1) between specific MAC7100 family microcontroller mask sets and the functional descriptions contained in the *MAC7100 Microcontroller Family Reference Manual* (MAC7100RM). Refer to http://www.freescale.com for the latest updates.

1 Introduction

This errata provides information applicable to the following MCU mask set devices:

- 0L38Y mask of MAC7106, MAC7116, MAC7126, MAC7136
- 1L38Y mask of MAC7106, MAC7116, MAC7126, MAC7136

When contacting a Motorola representative for assistance, please have the MCU device mask set and date code information (described below) available.

1.1 MCU Device Part Number Prefixes

All MAC7100 family devices are marked with a PAC, MAC or SAC prefix. These prefixes denote the following:

PAC	Devices that have been tested, but are not fully characterized or qualified over the full range of normal manufacturing process variations.
MAC	Fully characterized and qualified standard devices.
SAC	Fully characterized and qualified special or custom devices.

This document contains information on a new product. Specifications and information herein are subject to change without notice.

1.2 MCU Device Mask Set Identification

The mask set of a device is identified by a four-character code consisting of a letter, two numerical digits, and a letter, for example L38Y. Slight variations to the mask set identification code may result in an optional numerical digit preceding the standard four-character code, for example 0L38Y.

1.3 MCU Device Date Codes

In addition to the part number and mask set markings, each device is marked to indicate the week of manufacture. The date is coded as four numerical digits where the first two digits indicate the year and the last two digits indicate the work week. For example, the date code “0412” indicates that the device was manufactured during the 12th week of the year 2004.

1.4 Errata System Tracking Numbers

MUCts0*xxxx* is the tracking number for MAC7100 family device errata. An errata number can be used with the mask set and date code to identify a specific errata to a Motorola representative.

2 Errata Summary

Table 1. Summary of MAC71*x*6 Mask Set Errata

Errata Number	Brief Description	Module(s) Affected	Mask Set Affected	
			0L38Y	1L38Y
MUCts01793	IPM/IPWM Modes, Value Read From UCAn May Be Incorrect	eMIOS	Yes	Yes
MUCts01831	Deactivating A Receive MB May Corrupt Another Active Receive MB	FlexCAN	Yes	Yes
MUCts01856	Changing CTARs Between Frames in Continuous PCS Mode May Cause Error	DSPI	Yes	Yes
MUCts01915	Clock Monitor Reset Causes System Lock-up	CRG	Yes	Yes
MUCts01916	VREG High Temperature Control Register (VREGHTCL) For Factory Use Only	VREG	Yes	Yes
MUCts01917	eMIOS Buffered Modes Not Available	eMIOS	Yes	Yes
MUCts02084	MCM Reset Status Register (MRSR) Always Reads 0x80	MCM	Yes	Yes
MUCts02468	Flash Controller Returns Incorrect Data On Certain Read Accesses	CFM	Yes	—
MUCts02523	FlexCAN Transmit Buffers May Freeze or Indicate Missing Frame	FlexCAN	Yes	Yes
MUCts02527	Debug Status Port Mode 2 Incorrect Signal Assignments in Documentation	SSM	Yes	Yes

3 Errata Details

This section provides a detailed description of each errata and a description of a possible work-around, where appropriate.

3.1 MUCts01793 — IPM/IPWM Modes, Value Read From UCA*n* May Be Incorrect

Description

When reading the UCA*n* register in Input Pulse Width Measurement (IPWM) or Input Period Measurement (IPM) modes, if the IPS bus cycle starts on the same clock cycle as an A2 capture, the data read will not be coherent with the one at the next UCB*n* read.

- In IPWM mode, data read from UCB*n* will be greater than UCA*n* (UCB*n* minus UCA*n* will be the pulse width measurement of the polarity opposite that defined by EDPOL).
- In IPM mode, data read from UCA*n* and UCB*n* will be the same.

The expected scenario is that UCA*n* will be greater than UCB*n* for both modes. Note that coherency is guaranteed in a sequence of several measurements only if the combined UCA*n* / UCB*n* reads for each new measurement are performed after the correspondent new flag event.

Work-Around

After reading UCA*n* and UCB*n*, if UCA*n* is not greater than UCB*n*, discard this pulse measurement and read both registers again in the usual order: first read UCA*n*, then read UCB*n*.

3.2 MUCts01831 — Deactivating A Receive MB May Corrupt Another Active Receive MB

Description

Deactivating a FlexCAN receive message buffer (MB) may cause corruption of another active receive MB if the following sequence occurs:

1. A receive MB is locked via reading the Control/Status word, and has a pending message in the temporary receive serial message buffer (SMB).
2. A message is received that matches a second receive MB, and is queued in the second SMB.
3. The first MB is unlocked during the time between the CRC field and the 6th bit of EOF.
4. The second MB is deactivated within 9 f_{IPS} clock cycles of the first MB being unlocked, resulting in corruption of the first MB.

Work-Around

Do not write to the Control/Status word after initializing a receive MB, and use the IFLAG status bit to determine reception of a new frame, as the Control/Status field will always indicate FULL or OVERRUN after receiving the first frame.

If a write (deactivation) is required to the Control/Status field of an active receive MB, a delay of 25 CAN bit times plus 9 f_{IPS} clock cycles between unlocking one MB and deactivating another MB will avoid corruption, however frames may still be lost.

3.3 MUCts01856 — Changing CTARs Between Frames in Continuous PCS Mode May Cause Error

Description

Under some conditions in continuous operation mode (CONT = 1), the command word associated with the data frame is not always executed properly. An incorrect transfer may occur when multiple frames are transferred in continuous PCS mode and the frames use different CTAR registers. For example, if an application tries to transmit a 12-bit frame and a 16-bit frame without negating PCS, two 12-bit frames are transferred. This has been observed in simulations where CPHA = 0. The two frames are transmitted correctly if CPHA = 1.

Work-Around

When CPHA = 0 and continuous PCS mode is used, extended length (> 16 bits) frames may be created only by using two frames of equal size. This means that certain frame sizes cannot be constructed (prime numbers > 16).

3.4 MUCts01915 — Clock Monitor Reset Causes System Lock-up

Description

If the clock monitor reset function is enabled and a clock monitor time-out occurs, the chip will be placed into a reset state, but it will never exit that state. In order to trigger this errata all of the following conditions must be true:

- Clock monitor is enabled (CRG PLLCTL[CME] = 0b1)
- Loss of clock is detected
- Self-clock mode is disabled (CRG PLLCTL[SCME] = 0b0)

The only way to recover from this error is via an external or low-voltage reset sequence.

Work-Around

The default values of the CRG PLLCTL[CME] and PLLCTL[SCME] bits are 0b1, and thus the error is avoided if the clock monitor reset mode is never enabled. In order to prevent inadvertent enabling of the clock monitor reset function, PLLCTL[SCME] should be written with 0b1 during the first write to the PLLCTL register following a reset. Since the PLLCTL[SCME] bit is write once, this will prevent the enabling of clock monitor reset.

There is no plan to correct this function, as will be reflected in future revisions of the *MAC7100 Microcontroller Family Reference Manual* (MAC7100RM).

3.5 MUCts01916 — VREG High Temperature Control Register (VREGHTCL) For Factory Use Only

Description

Although information describing the VREGHTCL register is included in the *MAC7100 Microcontroller Family Reference Manual* (MAC7100RM) revisions 0.6, 0.6.1 and 1.0, the functions provided by this register are not fully characterized for customer use. Thus, VREGHTCL is reserved for factory testing during manufacturing processes, and is not suitable for application use.

Work-Around

There is no work-around available, nor are there plans to characterize this circuitry for customer use. Future versions of the MAC7100RM will remove the descriptions of the high temperature functions.

3.6 MUCts01917 — eMIOS Buffered Modes Not Available

Description

Table 20-9 and Sections 20.6.7.15 through 20.6.7.18 of the *MAC7100 Microcontroller Family Reference Manual* (MAC7100RM) revision 1.0, describe four buffered UC modes which are implemented on other Freescale devices, but are not available on any MAC7100 family devices. The described, unavailable UC modes are:

- Modulus Counter, Buffered (MCB)
- Output Pulse Width and Frequency Modulation, Buffered (OPWFMB)
- Center Aligned Output Pulse Width Modulation, Buffered (OPWMCB)
- Output Pulse Width Modulation, Buffered (OPWMB)

Work-Around

The UCCR*n*[MODE] field should not be set to `0b1010000` through `0b1100010` (refer to Table 20-9). Future revisions of the *MAC7100 Microcontroller Family Reference Manual* (MAC7100RM) will remove descriptions of these UC modes.

3.7 MUCts02084 — MCM Reset Status Register (MRSR) Always Reads 0x80

Description

The reset status register (MRSR) in the MCM module, which is intended to provide a read-only status of the last reset event, returns an incorrect status. In particular, the functionality of the MRSR reporting is limited such that all reset events are reported as power-on resets, regardless of the actual source of the reset. As a result, all reads of the MRSR return a data value of 0x80.

Work-Around

There is no work-around available.

3.8 MUCts02468 — Flash Controller Returns Incorrect Data On Certain Read Accesses

Description

On MAC71*x*6 devices, the Flash requires two cycles to return data on a read. For improved instruction execution performance, the Flash controller performs speculative reads such that in many cases the Flash has begun fetching the data before it is requested. Due to this mechanism the read data can be returned to the CPU with no wait states.

In cases where a new read request arrives for the upper half (`0x0008_0000` to `0x000F_FFFF`) while a speculative read is in progress to the lower half (`0x0000_0000` to `0x0007_FFFF`), the speculative access is not properly cancelled and the lower half data may be returned instead of the expected data.

Work-Around

Avoiding all direct references to the upper half of the Flash will avoid the lower-to-upper transition. The upper half can be accessed via the IPS interface (`0xFC18_0000` to `0xFC1F_FFFF`), however this has a significant performance impact, as read accesses from this address range can take ten times the number of clock cycles to execute as accesses to the direct address range (`0x0008_0000` to `0x000F_FFFF`).

3.9 MUCts02523 — FlexCAN Transmit Buffers May Freeze or Indicate Missing Frame

Description

If a received frame is serviced during reception of a second frame identified for that same MB (message buffer) and a new Tx frame is also initiated during this time, the Tx MB can become frozen and will not transmit while the bus is idle. The MB remains frozen until a new frame appears on the bus.

If the new frame is a received frame, the frozen MB is released and will arbitrate for external transmission. If the new frame is a transmitted frame from another Tx MB, the frozen MB changes its C/S (control status word) and IFLAG to indicate that transmission has occurred, although no frame was actually transmitted.

The frozen MB occurs if lock, unlock and initiate Tx events all occur at specific times during reception of two frames. The timing of the lock event affects the timing window of the unlock event as follows:

Situation A: Rx MB is locked during the second frame.

A frozen Tx MB occurs if:

1. Both of these events occur in either a-then-b or b-then-a order:
   a) A new transmission is initiated by writing its C/S sometime between CRC3 (third bit of CRC field) and EOF7 (seventh bit of end of frame) of the second frame.
   b) The Rx MB is locked by reading its C/S word sometime after EOF6 of first frame and before EOF6 of second frame.
2. The Rx MB is unlocked between EOF7 and intermission at end of the second frame.

Notice in this situation that if the lock / unlock combination happens close together, the lock must have been just before EOF6 of the second frame, and therefore the system is very close to having an overrun condition due to the delayed handling of received frames.

Situation B: Rx MB was locked before EOF6 of the first frame; in other words, before its IFLAG is set.

This is a less likely situation but provides a larger window for the unlock event. A frozen Tx MB occurs if:

1. The Rx MB is locked by reading its C/S word before EOF6 of the first frame.
2. Both of these events occur in either a-then-b or b-then-a order:
   a) A new transmission is initiated by writing its C/S word sometime between CRC3 and EOF7 of the second frame.
   b) The Rx MB is unlocked between CRC3 and intermission at end of the second frame.

Notice in this situation that if the unlock occurs after EOF6, the first frame would be lost and the second frame would be moved to the Rx MB due to the delayed handling of received frames.

Situation C: Rx unlocked during bus idle.

A frozen/missing Tx occurs if:

1. An Rx MB is locked before EOF6 of an incoming frame with matching ID and remains locked at least until intermission. This situation would usually occur only if the received frame was serviced after reception of a second frame.
2. An internal arbitration period is triggered by writing a C/S field of an MB.
3. The locked Rx MB is unlocked within two internal arbitration periods (defined below) of step 2.
4. 0xC is written to the C/S field of a Tx MB within these same two arbitration periods. This step is optional if a 0xC was written in step 2 above.

Two internal arbitration periods are calculated as:

$$\frac{(2 \times \text{number of MBs}) + 16}{f_{\text{IPS}}} = t_{\text{ARB}} \qquad \textit{Eqn. 1}$$

The number of MBs can be reduced by writing to the FlexCAN MCR[MAXMB] field. The f_{IPS} clock frequency is used in this calculation regardless of the CTRL[CLK_SRC] setting.

Additional Notes:

- The received frames can be transmitted from the same node, but they must be received into an Rx MB.
- When the frozen Tx MB's IFLAG becomes set, an interrupt will occur if enabled.
- The timestamp of the missing Tx will be set to the same timestamp value as the last reception before it was frozen.
- If the user software locks the Rx MB before a frame is received, situation A can occur with a single received frame.
- The issue does not occur if there were any additional pending Tx MBs before CRC3.
- If multiple Tx MBs are initiated within the CRC3/EOF7 window (situation A and B) or two internal arbitration windows (situation C), they all become frozen.

Work-Around

If received frames can be handled (lock/unlocked) before EOF6 of the next frame, situations A and C are avoided. If they are handled before CRC3, or lock times are below 23 CAN bit times, situation B is avoided.

If these conditions cannot be guaranteed by the existing system design, situations A and B are avoided by inserting a delay of at least 28 CAN bit times between initiating a transmission and unlocking an Rx MB and vice versa. Typically a system will use a mechanism to selectively add the necessary delay. For example, software might use a global variable to record an external timer value (the FlexCAN timer can’t be used, as that would unlock) when initiating a new Tx or unlocking an Rx, and then add the required delay before performing the second action.

Situation C can also be avoided by inserting a delay of at least two internal arbitration periods between writing 0xC and unlocking the locked Rx MB.

3.10 MUCts02527 — Debug Status Port Mode 2 Incorrect Signal Assignments in Documentation

Description

The signal assignments listed for Debug Status Port Mode 2 in Chapter 26, “System Services Module (SSM),” of the *MAC7100 Microcontroller Family Reference Manual* (MAC7100RM) v 1.0 are incorrect. The table below shows the incorrect and correct signal assignments.

Table 2. Port F Debug Status Mode 2 Correct Signal Assignments

Port F Pin	Mode 2 Function	
	Incorrect	Correct
PF0	System is entering STOP mode	System is entering STOP mode
PF1	Platform has entered STOP mode	Platform has entered STOP mode
PF2	ATD A has entered STOP mode	ATD B has entered STOP mode
PF3	ATD B has entered STOP mode	ATD A has entered STOP mode
PF4	SCI A has entered STOP mode	PIT has entered STOP mode
PF5	SCI B has entered STOP mode	I^{2}C has entered STOP mode
PF6	SCI C has entered STOP mode	CAN D has entered STOP mode
PF7	SCI D has entered STOP mode	CAN C has entered STOP mode
PF8	CAN A has entered STOP mode	CAN B has entered STOP mode
PF9	CAN B has entered STOP mode	CAN A has entered STOP mode
PF10	CAN C has entered STOP mode	SPI B has entered STOP mode
PF11	CAN D has entered STOP mode	SPI A has entered STOP mode
PF12	PIT has entered STOP mode	SCI D has entered STOP mode
PF13	I^{2}C has entered STOP mode	SCI C has entered STOP mode
PF14	SPI A has entered STOP mode	SCI B has entered STOP mode
PF15	SPI B has entered STOP mode	SCI A has entered STOP mode

Work-Around

Use the correct signals as shown above.

THIS PAGE INTENTIONALLY LEFT BLANK

THIS PAGE INTENTIONALLY LEFT BLANK

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

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)
support@freescale.com

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 Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Information in this document is provided solely to enable system and software implementers to use Freescale Semiconductor products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits or integrated circuits based on the information in this document.

Freescale Semiconductor reserves the right to make changes without further notice to any products herein. Freescale Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in Freescale Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals”, must be validated for each customer application by customer’s technical experts. Freescale Semiconductor does not convey any license under its patent rights nor the rights of others. Freescale Semiconductor products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the Freescale Semiconductor product could create a situation where personal injury or death may occur. Should Buyer purchase or use Freescale Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold Freescale Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Freescale Semiconductor was negligent regarding the design or manufacture of the part.

freescale™
semiconductor