Mask Set Errata for Mask 3M05C

Introduction

This report applies to mask 3M05C for these products:

- MC9S08DZ60
- MC9S08DZ48
- MC9S08DZ32
- MC9S08DZ16
- MC9S08DV60
- MC9S08DV48
- MC9S08DV32
- MC9S08DV16
- MC9S08DN60
- MC9S08DN48
- MC9S08DN32
- MC9S08DN16
- MC9S08DE60
- MC9S08DE32
- MC9S08EN32
- MC9S08EN16

MCU device mask set identification

The mask set is identified by a 5-character code consisting of a version number, a letter, two numerical digits, and a letter, for example 0J27F. All standard devices are marked with a mask set number and a date code.

MCU device date codes

Device markings indicate the week of manufacture and the mask set used. 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 instance, the date code "0301" indicates the first week of the year 2003.

freescale
semiconductor

MCU device part number prefixes

Some MCU samples and devices are marked with an SC, PC, or XC prefix. An SC prefix denotes special/custom device. A PC prefix indicates a prototype device which has undergone basic testing only. An XC prefix denotes that the device is tested but is not fully characterized or qualified over the full range of normal manufacturing process variations. After full characterization and qualification, devices will be marked with the MC or SC prefix.

SE156-ADC-COCO: COCO bit may not get cleared when ADCSC1 is written to

Errata type:	Silicon
Affected component:	ADC
Description:	If an ADC conversion is near completion when the ADC Status and Control 1 Register (ADCSC1) is written to (i.e., to change channels), it is possible for the conversion to complete, setting the COCO bit, before the write instruction is fully executed. In this scenario, the write may not clear the COCO bit, and the data in the ADC Result register (ADCR) will be that of the recently completed conversion.

If interrupts are enabled, then the interrupt vector will be taken immediately following the write to the ADCSC1 register. |
| **Workaround:** | It is recommended when writing to the ADCSC1 to change channels or stop continuous conversion, that you write to the register twice. The first time should be to turn the ADC off and disable interrupts, and the second should be to select the mode/channel and re-enable the interrupts. |

SE157-ADC-INCORRECT-DATA: Boundary case may result in incorrect data being read in 10- and 12-bit modes

Errata type:	Silicon
Affected component:	ADC
Description:	In normal 10-bit or 12-bit operation of the ADC, the coherency mechanism will freeze the conversion data such that when the high byte of data is read, the low byte of data is frozen, ensuring that the high and low bytes represent result data from the same conversion.

In the errata case, there is a single-cylce (bus clock) window per conversion cycle when a high byte may be read on the same cycle that subsequent a conversion is completing. Although extremely rare due to the precise timing required, in this case, it is possible that the data transfer occurs, and the low byte read may be from the most recently completed conversion.

In systems where the ADC is running off the bus clock, and the data is read immediately upon completion of the conversion, the errata will not occur. Also, in single conversion mode, if the data is read prior to starting a new conversion, then the errata will not occur.

The errata does not impact 8-bit operation.

Introducing significant delay between the conversion completion and reading the data, while a following conversion is executing/pending, could increase the probability for the errata to occur. Nested interrupts, significant differences |

Mask Set Errata for Mask 3M05C, Rev. 0, 7/2008

Freescale Semiconductor

between the bus clock and the ADC clock , and not handling the result register reads consecutively, can increase the delay and therefore the probability of the errata occuring.

Workaround: Using the device in 8-bit mode will eliminate the possibility of the errata occuring.

Using the ADC in single conversion mode, and reading the data register prior to initiating a subsequent conversion will eliminate the possibility of the errata occuring.

Minimizing the delay between conversion complete and processing the data can minimize the risk of the errata occuring. Disabling interrupts on higher priority modules and avoiding nested interrupts can reduce possible contentions that may delay the time from completing a conversion and handling the data. Additionally, increasing the bus frequency when running the ADC off the asynchronous clock, may reduce the delay from conversion complete to handling of the data.

SE143-ICS: ICS Internal Reference Can Remain Enabled in Stop3 Mode

Errata type: Silicon

Affected component: ICS

Description: When transitioning from FEI or FBI modes to FEE or FBE modes, the internal reference clock may remain active in stop3 mode if the STOP instruction is executed soon after the IREFST bit in the ICSSC register clears. This can lead to elevated stop3 I_{DD}.

Workaround: To ensure the internal reference clock is disabled before entering stop3, wait three internal reference clock periods after the IREFST bit has cleared before entering stop3. On a device with a trimmed internal reference, one period will be between 25.6 µs and 32 µs, therefore waiting 100 µs is adequate for all trimmed devices.

Or

To ensure the internal reference clock is disabled before entering stop3, transition into FEE mode and wait until the LOCK status bit indicates the FLL has attained lock before entering stop3 or transitioning into FBE mode and entering stop3.

SE128C-MCGV1: MCGV1 Can Cause a Very Short Clock Pulse

Errata type: Silicon

Affected component: MCGV1

Description: The MCG module V1 — when configured with the FLL enabled and with BDIV set to divide-by one — can sometimes produce a very short clock pulse. This short clock pulse can cause the device to malfunction. The short clock pulse is caused when the digitally controlled oscillator (DCO) crosses a filter value boundary when compensating for output frequency error. The filter value is not in the memory map and cannot be read by user code.

Mask Set Errata for Mask 3M05C, Rev. 0, 7/2008

- When operating from the internal reference clock, certain trim values can cause the error more often. The trim value for any particular clock frequency is unique to each device.
- The temperature coefficient of the DCO is such that the unique reference frequency causing the error, either internally or externally generated, will not be constant over temperature.

Workaround:
- If using FLL enabled with internal reference (FEI) or FLL enabled with external reference (FEE) modes, operate the device with bus frequency below 10 MHz. This is accomplished by setting BDIV divide-by value to two or higher (BDIV[1,0] bit field value of 01, 10 or 11).
- Use the MCG in any mode in any mode other than FEI or FEE. Options include: FLL bypassed internal (FBI), FLL bypassed external (FBE), PLL engaged external (PEE), PLL bypassed external (PBE), bypassed low power internal (BLPI) and bypassed low power external (BLPE). the BLPE mode.

SE127-IIC: IIC Does Not Function When High Drive Strength Is Enabled on the IIC Pins

Errata type: Silicon

Affected component: PWM

Description: IIC operation requires open drain configuration but when the PTxDS bit associated with the IIC SDA or SCL pin is set for high drive strength (PTxDS = 1) the pin will be in complementary drive mode instead of open drain mode. This will cause IIC communication conflicts that lead to IIC malfunction.

Workaround: The PTxDS bits associated with the IIC SDA and SCL pins should be cleared for low drive strength operation. This configuration will ensure that the corresponding pin will enter open drain mode and IIC communications will function correctly. The IIC pull-up resistor in the final application may need to be modified to account for the lower drive strength of the SDA and SCL pins.

SE120-IIC: Incorrect Clocking in 10-Bit Addressing Mode

Errata type: Silicon

Affected component: IIC

Description: This erratum is relevant only for applications using 10-bit addressing mode and does not affect 7-bit addressing mode operations.

When hexidecimal values E0,E1,E4,E5,E8,E9,EC,ED,F0,F2,F4, or F6 are used as data in master receive slave transmissions from the IIC module using 10-bit addresses, the IIC clock may produce an incorrect number of pulses. This will result in IIC communication errors.

Workaround: Avoid using the values listed in the erratum description as data in any master receive slave transmission from the IIC module while in 10-bit addressing mode.

Mask Set Errata for Mask 3M05C, Rev. 0, 7/2008

SE118-RTC: LPO Enabled in Stop Modes When RTC is in Reset State

Errata type: Silicon

Affected component: RTC

Description: The RTC module comes out of reset with the low power oscillator (LPO) as the selected clock source (RTCLKS = 0:0), but with the RTC module itself disabled (RTCPS = 0:0:0:0). This configuration signals the LPO to be enabled in all modes including stop2 and stop3.

     This results in higher than expected stop currents due to the LPO stop mode current adder.

Workaround: To disable the LPO when not needed in the stop modes, set the RTCLKS bits in the RTCSC register to any value other than 0:0. This will select an alternative clock source for the RTC, either the external clock (ERCLK) or the internal clock (IRCLK). Unlike the LPO, the RTC module does not automatically enable either of these clocks to run in stop modes. ERCLK or IRCLK enable is determined by other MCU control bits, not the RTC's. Before entering stop mode, one period of the LPO must expire after setting the RTCLKS bits to ensure that the change has been processed.

SE110-TPM: PWM Boundary Case Issues in HCS08 Timer PWM Module (TPM)

Errata type: Silicon

Affected component: PWM

Description: This errata describes boundary case issues that primarily affect the center-aligned PWM mode of operation. While investigating these issues, additional, less significant, issues were discovered. These will be explained, although they should not cause any significant problems in normal applications.

     In center-aligned PWM mode, the timer counter counts up until it reaches the modulo value in TPMMODH:TPMMODL, reverses direction, and then counts down until it reaches zero, where it reverses and counts up again. A period of the PWM output is centered around the leading edge of the zero count and the period is considered to start when the count changes from TPMMODH:TPMMODL–1 to TPMMODH:TPMMODL (the same point where the counter changes from up-counting to down-counting). The zero value and the maximum modulo value occur for only one timer count cycle each, while all other values occur twice (once during the down-counting phase and again during the up-counting phase). Therefore, the total period of the PWM signal is two times the value in TPMMODH:TPMMODL.

     The value on each TPM timer output pin is controlled by an internal flip-flop that is cleared at reset but is not readable by software. These internal flip-flops change state when timer output compare events or PWM duty cycle compare events occur (when the channel value registers match the timer count registers). This leads to these outputs remaining in a previous state until a compare event occurs after changing the configuration of the timer system. When the timer is initialized the first time after a reset, the state of these output flip-flops is known to be reset (logic low). If the configuration is changed after the channel has been running in another configuration for some period

of time, you sometimes do not know the state of these internal flip-flops (and therefore the state of the timer output pins) until a new channel value register compare event occurs. There is nothing improper about these periods before the first event occurs, however some users might be surprised the first time they notice this behavior.

When the MCU is reset, the count (TPMCNTH:TPMCNTL) is reset to 0x0000. If the timer is configured for center-aligned pulse-width modulation (PWM) and then the clock is started, this corresponds to the middle of a PWM period. If the internal flip-flop corresponding to the output was at the inactive level when the PWM started, this would appear as if there was an extra half period of delay before the first full PWM cycle started. If the internal flip-flop corresponding to the output happened to be at the active level when this PWM was started, a pulse equivalent to half of a normal duty cycle pulse could be produced at the PWM output pin.

There are eight cases discussed in this errata:

- Cases 1 and 2 — These are two error cases near the 100% duty cycle boundary. The first is when the channel value registers are set equal to the modulo value. The second is when the channel value registers are set to one less than the modulo value.
- Cases 3, 4, and 5 — These cases are related to changing the channel value to or from 0x0000. The errors depend upon whether this is done during the first or second half of the center-aligned PWM period. In all of these cases, the workaround strategy is to produce 0% duty cycle with a negative channel value instead of using the 0x0000 value. This can be done by checking any value that is about to be written to the channel value registers, and then decrement the 16-bit value or the high-order byte of the value before writing it to the channel value registers. This produces the desired 0% duty cycle and avoids the problems related to a zero in the channel value registers.
- Case 6 — Although this behavior wasn't discussed in the data sheet, the operation is different than some users might expect. In edge-aligned PWM mode, when the channel value is changed from zero to a non-zero value, the new PWM settings can take an extra half PWM period to take effect. It is unlikely that this would cause any problems in any practical application system.
- Case 7 — This case is more of a clarification of an unusual situation rather than a design problem. This case happens when the prescale factor is changed during operation and only affects center-aligned PWM. It would be very unusual to change the prescale setting after it is set during reset initialization. The prescale flip-flops are not reset when the prescale setting is changed, so the first prescaled clock period after a change may be shorter or longer than expected.
- Case 8 — This case would only arise when a series of unlikely events happened to occur. It affects only center-aligned PWM mode if the timer counter is stopped, reset, and restarted when the count value happened to be equal to the TPMxMODH:TPMxMODL value. Because the timer counter would not normally be stopped during operation in center-aligned PWM mode, this case should never arise in a practical application.

Mask Set Errata for Mask 3M05C, Rev. 0, 7/2008

Workaround:

Case 1: Center-Aligned PWMChannel Value (TPMxCnVH:TPMxCnVL) = Modulo Value (TPMxMODH:TPMxMODL)

This should produce 100% duty cycle where the TPM output pin remains at the active level continuously. Instead, the output remains at the inactive level, which corresponds to 0% duty cycle.

Check any value that is about to be written to the channel value registers. If the value is the same as the modulo value, increment the value before writing it to the channel value register. This workaround will work for any modulo value that is greater than zero and less than 0x7FFF. Setting the channel value to any 2's complement negative value (0x8000 through 0xFFFF) results in 0% duty cycle as expected and described in the original TPM documentation.

Another workaround would be to choose not to use 100% duty cycle in the application. Not all applications require the range to include the 100% duty cycle case.

Case 2: Center-Aligned PWMChannel Value (TPMxCnVH:TPMxCnVL) = Modulo Value Minus 1 (TPMxMODH:TPMxMODL – 1)

This should produce almost 100% duty cycle where the TPM output pin remains at the active level for [(TPMxCnVH:TPMxCnVL × 100)/(TPMxMODH:TPMxMODL)]% of the period. Instead, the output remains at the inactive level which corresponds to 0% duty cycle.

Reduce the prescale factor by a factor of two and then multiply the modulo and channel value settings by a factor of two. In this way, the frequency and resolution of the PWM output remain the same but channel values are always even numbers and are never equal to the modulo setting minus one.

Consider the case of a 20-MHz bus frequency, 25-kHz PWM frequency, and 0.25% resolution on the duty cycle. Before making the adjustments suggested in this workaround, you could have the following setup: Set the modulo to 400 and the prescale factor in PS2:PS1:PS0 to divide by 2 (0:0:1). Each step of the channel value from 0–1–2…398–399–400 would increase the duty cycle by 0.25%.

Increasing the modulo value to 800 and reducing the prescale factor to divide-by one, would still produce the same period or PWM frequency. If the original channel values were multiplied by two (shift left one bit position) before writing them to the channel value register, the resolution would still be 0.25% per step of the channel value, but the channel values would step by 2 each time as in 0-2-4-6…796-798-800. With this workaround, the channel value would never be equal to the modulo value minus one, and the error condition would not arise.

With common HCS08 bus frequencies, practical PWM frequencies, and reasonable resolution requirements, there is enough speed and flexibility in the TPM system so this workaround should work well with all except the most unusual application systems.

Another workaround would be to limit the range of allowed values in the channel value register so it does not include the TPMxMODH:TPMxMODL or (TPMxMODH:TPMxMODL – 1) values. Not all applications require the range to include these values.

Case 3: Center-Aligned PWMTPMxCnVH:TPMxCnVL Changed from 0x0000 to a Non-Zero Value

Mask Set Errata for Mask 3M05C, Rev. 0, 7/2008

This case occurs only while the counter is counting down (first half of the center-aligned PWM period). The PWM output changes to the active level at the middle of the current PWM period as the count reaches 0x0000 instead of waiting for the start of a new PWM period to begin using the new duty cycle setting.

Use a negative channel value instead of 0x0000 to produce 0% duty cycle. This can be done by checking any value that is about to be written to the channel value registers, and then decrementing the 16-bit value or the high-order byte of this value before writing it to the channel value registers. This produces the desired 0% duty cycle and it avoids the problems related to a zero in the channel value registers.

Case 4: Center-Aligned PWMTPMxCnVH:TPMxCnVL Changed from a Non-Zero Value to 0x0000

This case occurs only while the counter is counting up (second half of the center-aligned PWM period) but before the count reaches the channel value setting in TPMxCnVH:TPMxCnVL. The PWM output remains at the active level until the end of the current PWM period instead of finishing the current PWM period using the old channel value setting.

Use a negative channel value instead of 0x0000 to produce 0% duty cycle. This can be done by checking any value that is about to be written to the channel value registers, and then decrement the 16-bit value or the high-order byte of this value before writing it to the channel value registers. This produces the desired 0% duty cycle and it avoids the problems related to a zero in the channel value registers.

Case 5: Center-Aligned PWMTPMxCnVH:TPMxCnVL Changed from 0x0000 to a Non-Zero Value

This case occurs only while the counter is counting down (first half of the center-aligned PWM period) and then TPMxCnVH:TPMxCnVL is changed back to 0x0000 during the first half of the next PWM period (while the counter is counting down). This is a very unlikely case in any practical application. The PWM output changes to the active level at the middle of the first PWM period as the count reaches 0x0000 instead of waiting for the start of a new PWM period to begin using the new duty cycle setting, and then the output remains active until the end of the second PWM period. In this very unusual case, the PWM output remains active for one and one-half PWM periods rather than remaining inactive for the first PWM period and then active for 2 × TPMxCnVH:TPMxCnVL during the next PWM period.

Use a negative channel value instead of 0x0000 to produce 0% duty cycle. This can be done by checking any value that is about to be written to the channel value registers, and then decrementing the 16-bit value or the high-order byte of this value before writing it to the channel value registers. This produces the desired 0% duty cycle and it avoids the problems related to a zero in the channel value registers.

Case 6: Edge-Aligned PWMTPMxCnVH:TPMxCnVL Changed from 0x0000 to a Non-Zero Value

This is a minor issue related to edge-aligned PWM when duty cycle is changed from 0x0000 to a non-zero value. This issue is a specification clarification rather than a design error.

Mask Set Errata for Mask 3M05C, Rev. 0, 7/2008

Freescale Semiconductor

In this case, the channel value update occurs at the same time as the new PWM period begins, but due to circuit delays, the update occurs slightly too late for the new duty cycle to take effect for that PWM period and an extra period of 0% duty cycle is produced. This causes the new PWM duty cycle to take effect one PWM period later than expected. This should not cause any application problems so the data book functional description will be changed to clarify this situation.

Case 7: Changing the Counter Prescaler while the TPM Counter Is Disabled

This case would not arise in most applications because it would be unusual to change the prescaler at any time other than initial timer setup after reset.

1. TPM counter was previously running
2. Counting is stopped by writing 0:0 to CLKS[1:0]
3. Change prescale value PS[2:1:0] to a different value while keeping clocks off (CLKS[1:0] = 0:0)
4. Clear the counter by writing any value to TPMxCNTH:TPMxCNTL
5. Turn clocks back on by writing to CLKS[1:0]

Unexpected Operation: The prescaler divider flip-flops begin counting from the prior value rather than starting from zero. This can result in the counter detecting the first clock edge after restarting, either earlier or later than expected.

Case 8: Center-Aligned PWM, Counter is Stopped, Reset, and Restarted when Counting Up and Count Equals the Modulo Value

This case is extremely unlikely to occur in any practical application because it would be very unusual to stop or reset the TPM counter while using center-aligned PWM mode.

1. TPM counter is counting up in center-aligned PWM mode (second half of a PWM period)
2. Counter is stopped (write CLKS[1:0] = 0:0) when count equals modulo value (the direction would normally change from up counting to down counting at the next clock edge)
3. Counter is reset to 0x0000 by writing any value to TPMxCNTH:TPMxCNTL
4. Counter is turned on again by writing to CLKS[1:0]

Unexpected Operation: Because the internal up/down indicator was not cleared when the counter was reset, the counter begins counting down from 0x0000 to 0xFFFF–0xFFFE… This causes the timing of the first PWM period after the counter reset to be longer than expected.

Mask Set Errata for Mask 3M05C, Rev. 0, 7/2008

How to Reach Us:

Home Page:
www.freescale.com

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

USA/Europe or Locations Not Listed:
Freescale Semiconductor
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

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