Mask Set Errata for Mask 0N66N_0N72D

This report applies to mask 0N66N_0N72D for these products:
- MPC567xK

Table 1. Errata and Information Summary

Erratum ID	Erratum Title
e3866	ADC Self Test algorithm S0 result can be incorrect at low temperature
e5569	ADC: The channel sequence order will be corrupted when a new normal conversion chain is started prior to completion of a pending normal conversion chain
e3416	DMA: Improper operation when used with Cache Coherency Unit.
e4396	e200z7: Erroneous Address Fetch
e6967	eDMA: Possible misbehavior of a preempted channel when using continuous link mode
e7877	FlexPWM: do not enable the fault filter
e8770	FlexRAY: Missing TX frames on Channel B when in dual channel mode and Channel A is disabled
e7589	LINFlexD: Spurious timeout error when switching from UART to LIN mode or when resetting LINTCSR[MODE] bit in LIN mode
e7394	MC_ME: Incorrect mode may be entered on low-power mode exit.
e5008	PMC: Potential stuck in reset condition if certain pins are pulled up too hard during power up.

Table 2. Revision History

Revision	Changes
15JUL2015	Initial release
31MAR2016	The following errata were added. • e3416

e3866: ADC Self Test algorithm S0 result can be incorrect at low temperature

Description: The ADC Self Test algorithm S step 0 (S0) measures ADC Vbandgap / VDD_HV_ADR. In the case where the S0 step occurs after the ADC has sampled and converted a value close to VDD_HV_ADR at low temperature (for example -40C) in some process corners the sampling time (specified by INPSAMP_S) of 80h is not long enough to allow the correct ADC sampling capacitor settling. This may lead to an incorrect converted value.

The Band Gap in the above specified condition is slow in discharging the sampling capacitor when the previous sampled voltage is much larger than the Band Gap output voltage (nominally 1.2V). The larger the voltage sampled before S0, the slower is the settling.

This issue can also affect S1 algorithm results since S1 = VDD_HV_ADV / Vbandgap.

Workaround: To eliminate the problem it is mandatory to:

(a) increase the sampling time for S supply self test (INSAMP_S) from 80h to FFh and

(b) insert a sacrificial ADC conversion immediately before the S supply self test.

The user software must insert a single-shot S algorithm Step 0 conversion (also called sacrificial S0 conversion) before the normal S supply self test to achieve accurate sample capacitor settling. The user software must prohibit any other conversions between the sacrificial S0 conversion and normal S0 conversion of the S supply self test.

e5569: ADC: The channel sequence order will be corrupted when a new normal conversion chain is started prior to completion of a pending normal conversion chain

Description: If One shot mode is configured in the Main Configuration Register (MCR[MODE] = 0) the chained channels are automatically enabled in the Normal Conversion Mask Register 0 (NCMR0). If the programmer initiates a new chain normal conversion, by setting MCR[NSTART] = 0x1, before the previous chain conversion finishes, the new chained normal conversion will not follow the requested sequence of converted channels.

For example, if a chained normal conversion sequence includes three channels in following sequence: channel0, channel1 and channel2, the conversion sequence is started by MCR[NSTART] = 0x1. The software re-starts the next conversion sequence when MCR[NSTART] is set to 0x1 just before the current conversion sequence finishes.

The conversion sequence should be: channel0, channel1, channel2, channel0, channel1, channel2.

However, the conversion sequence observed will be: channel0, channel1, channel2, channel1, channel1, channel2. Channel0 is replaced by channel1 in the second chain conversion and channel1 is converted twice.

Workaround: Ensure a new conversion sequence is not started when a current conversion is ongoing. This can be ensured by issuing the new conversion setting MCR[NSTART] only when MSR[NSTART] = 0.

Note: MSR[NSTART] indicates the present status of conversion. MSR[NSTART] = 1 means that a conversion is ongoing and MSR[NSTART] = 0 means that the previous conversion is finished.

e3416:   DMA: Improper operation when used with Cache Coherency Unit.

Description: When the Cache Coherency Unit (CCU) is enabled to monitor DMA writes, the DMA can occasionally apply an extra destination offset (doff) to the destination address (daddr) in the Transfer Control Descriptor of a channel. This causes the DMA to write outside the bounds defined by the application in the TCD structure.

Workaround: Disable write monitoring of the DMA channels and either inhibit caching for regions of memory shared between the DMA and CPU cores or handle coherency manually in software. The CCU may still be enabled to manage cache coherency for memory shared between the two CPU cores only.

Alternatively, if it is possible to reprogram the TCD with no destination offset (doff=0) then the extra offset will have no effect when the error occurs. If the TCD can be re-defined such that each minor loop consists only of a single data transfer (nBytes = dSize) then destination offset can be set to zero and minor loop offset (mloff) used instead to advance the destination address. This requires enabling Minor Loop Mapping for all channels by setting DMA_x_MCR[EMLM] = 1. TCDs not writing to coherent memory are unaffected by this mode provided their minor loop byte counts (nBytes) are not greater than or equal to 1 GB (The minor loop mapping mode uses the two high order bits of the nBytes field to enable source and/or destination minor loop offset).

e4396:   e200z7: Erroneous Address Fetch

Description: Under certain conditions, if a static branch prediction and a dynamic return prediction (which uses the subroutine return address stack) occur simultaneously in the Branch Target Buffer (BTB), the e200z7 core can issue an errant fetch address to the memory system (instruction fetched from wrong address).

This can only happen when the static branch prediction is "taken" but the branch actually resolves to "not taken". If the branch resolves to taken, correct fetching occurs for this branch path and no issue is seen.

If Branch Unit Control and Status Register (BUCSR) Branch Prediction Control Static (BPRED) = 0b00, 0b01, or 0b10, then static branch prediction is configured as "taken". The issue can occur with these settings.

If BUSCR[BPRED] = 0b11, then static branch prediction is configured as "not taken". The issue does not occur with this setting.

Workaround: To prevent the issue from occurring, configure static branch prediction to "not taken" by setting the Branch Unit Control and Status Register (BUCSR) Branch Prediction Control Static (BPRED) to 0b11.

e6967:   eDMA: Possible misbehavior of a preempted channel when using continuous link mode

Description: When using Direct Memory Access (DMA) continuous link mode Control Register Continuous Link Mode (DMA_CR[CLM]) = 1) with a high priority channel linking to itself, if the high priority channel preempts a lower priority channel on the cycle before its last read/write sequence, the counters for the preempted channel (the lower priority channel) are corrupted. When the

Mask Set Errata for Mask 0N66N_0N72D, Rev. 31MAR2016

preempted channel is restored, it continues to transfer data past its "done" point (that is the byte transfer counter wraps past zero and it transfers more data than indicated by the byte transfer count (NBYTES)) instead of performing a single read/write sequence and retiring.

The preempting channel (the higher priority channel) will execute as expected.

Workaround: Disable continuous link mode (DMA_CR[CLM]=0) if a high priority channel is using minor loop channel linking to itself and preemption is enabled. The second activation of the preempting channel will experience the normal startup latency (one read/write sequence + startup) instead of the shortened latency (startup only) provided by continuous link mode.

e7877: FlexPWM: do not enable the fault filter

Description: Operation of the fault pin filter of the Flexible Pulse Width Modulation (FLEX_PWM) may be inconsistent if the Fault Filter is enabled, by setting the Filter Period greater than zero in the Fault Filter register (FFILT[FILT_PER] > 0). The fault filter flag may be set even though the pulse is shorter than the filter time.

Workaround: Do not enable the PWM fault pin filters. Disable the fault pin filters by setting the Fault Filter Period to 0 in the Fault Filter Register (FFILT[FILT_PER] = 0).

e8770: FlexRAY: Missing TX frames on Channel B when in dual channel mode and Channel A is disabled

Description: If the FlexRay module is configured in Dual Channel mode, by clearing the Single Channel Device Mode bit (SCM) of the Module Control register (FR_MCR[SCM]=0), and Channel A is disabled, by clearing the Channel A Enable bit (FR_MCR[CHA]=0) and Channel B is enabled, by setting the Channel B enable bit (FR_MCR[CHB]=1), there will be a missing transmit (TX) frame in adjacent minislots (even/odd combinations in Dynamic Segment) on Channel B for certain communication cycles. Which channel handles the Dynamic Segment or Static Segment TX message buffers (MBs) is controlled by the Channel Assignment bits (CHA, CHB) of the Message Buffer Cycle Counter Filter Register (FR_MBCCFRn). The internal Static Segment boundary indicator actually only uses the Channel A slot counter to identify the Static Segment boundary even if the module configures the Static Segment to Channel B (FR_MBCCFRn[CHA]=0 and FR_MBCCFRn[CHB]=1). This results in the Buffer Control Unit waiting for a corresponding data acknowledge signal for minislot:N in the Dynamic Segment and misses the required TX frame transmission within the immediate next minislot:N+1.

Workaround: 1. Configure the FlexRay module in Single Channel mode (FR_MCR[SCM]=1) and enable Channel B (FR_MCR[CHB]=1) and disable Channel A (FR_MCR[CHA]=0). In this mode the internal Channel A behaves as FlexRay Channel B. Note that in this mode only the internal channel A and the FlexRay Port A is used. So externally you must connect to FlexRay Port A.

2. Enable both Channel A and Channel B when in Dual Channel mode (FR_MCR[CHA=1] and FR_MCR[CHB]=1). This will allow all configured TX frames to be transmitted correctly on Channel B.

e7589: LINFlexD: Spurious timeout error when switching from UART to LIN mode or when resetting LINTCSR[MODE] bit in LIN mode

Description: If the LINFlexD module is enabled in Universal Asynchronous Receiver/Transmitter (UART) mode and the value of the MODE bit of the LIN Timeout Control Status register (LINTCSR) is 0 (default value after reset), any activity on the transmit or receive pins will cause an unwanted change in the value of the 8-bit field Output Compare Value 2 (OC2) of the LIN Output Compare register (LINOCR).

If the LINFlexD module is enabled in LIN mode and the value of the MODE bit of the LIN Timeout Control Status register (LINTCSR) is changed from '1' to '0', then the old value of the Output Compare Value 1 (OC1) and Output Compare Value 2 (OC2) of the LIN Output Compare register (LINOCR) is retained.

As a consequence, if the module is reconfigured from UART to Local Interconnect Network (LIN) mode, or LINTCSR MODE bit is changed from '1' to '0', an incorrect timeout exception is generated when the LIN communication starts.

Workaround: If the LINFlexD module needs to be switched from UART mode to LIN mode, before writing UARTCR[UART] to 1, ensure that the LINTCSR[MODE] is first set to 1.

If the LINFlexD module is in LIN mode and LINTCSR[MODE] needs to be switched from 1 to 0 in between frames, the LINOCR must be set to 0xFFFF by software.

e7394: MC_ME: Incorrect mode may be entered on low-power mode exit.

Description: For the case when the Mode Entry (MC_ME) module is transitioning from a run mode (RUN0/1/2/3) to a low power mode (HALT/STOP/STANDBY*) if a wake-up or interrupt is detected one clock cycle after the second write to the Mode Control (ME_MCTL) register, the MC_ME will exit to the mode previous to the run mode that initiated the low power mode transition.

Example correct operation DRUN->RUN1-> RUN3->STOP->RUN3

Example failing operation DRUN->RUN1-> RUN3->STOP->RUN1

*Note STANDBY mode is not available on all MPC56xx microcontrollers

Workaround: To ensure the application software returns to the run mode (RUN0/1/2/3) prior to the low power mode (HALT/STOP/STANDBY*) it is required that the RUNx mode prior to the low power mode is entered twice.

The following example code shows RUN3 mode entry prior to a low power mode transition.

ME.MCTL.R = 0x70005AF0; /* Enter RUN3 Mode & Key */

ME.MCTL.R = 0x7000A50F; /* Enter RUN3 Mode & Inverted Key */

while (ME.GS.B.S_MTRANS) {} /* Wait for RUN3 mode transition to complete */

ME.MCTL.R = 0x70005AF0; /* Enter RUN3 Mode & Key */

ME.MCTL.R = 0x7000A50F; /* Enter RUN3 Mode & Inverted Key */

while (ME.GS.B.S_MTRANS) {} /* Wait for RUN3 mode transition to complete */

/* Now that run mode has been entered twice can enter low power mode */

/* (HALT/STOP/STANDBY*) when desired. */

Mask Set Errata for Mask 0N66N_0N72D, Rev. 31MAR2016

e5008: PMC: Potential stuck in reset condition if certain pins are pulled up too hard during power up.

Description: If the below identified pins are pulled high either with a direct short or with too small a resistance during power up, a stuck in reset condition can occur due to a failure to deassert internal LVDs on the 3.3V rail or the 1.2V rail. This affects both internal and external regulation modes of the PMC. There is no issue if the pins are no connects/floating or pulled low with any resistor value.

The pins affected are:

Pin 1: ETIMER1_ETC5/SIUL_GPIO78/SIUL_EIRQ26 (Y15 for 473 BGA, P13 for 257 BGA)

Pin 2: ETIMER1_ETC3/SIUL_GPIO92/CTU1_EXT_IN/MC_RGM_FAB/SIUL_EIRQ30 (Y11 for 473 BGA, P8 for 257 BGA)

Pin 3: ETIMER1_ETC4/SIUL_GPIO93/CTU1_EXT_TGR/SIUL_EIRQ31 (Y16 for 473 BGA, P12 for 257 BGA)

Workaround: 1. If pins 1 or 3 need to be pulled high during power up, a 20 K ohm resistor or greater must be installed between each pin and the voltage source.

2. If the serial boot option is needed :

a) If entering during power up, a 20 K ohm 1% resistor must be installed between Pin 2 (the MC_RGM_FAB pin) and the voltage source and the ambient temperature is between -40C and 80C.

b) If entering after reset deassertion, the following sequence can be used from -40C to 125C.

i. Power up device with Pin 2 (MC_RGM_FAB pin) low.

ii. Verify device is out of reset.

iii. Pull FAB pin up to Vih level. This will be achieved with a resistance value less than 18 K ohm.

iv. Assert and deassert RESET_B either through a hardware induced pin toggle or a software induced destructive or functional reset using the ME module.

Document Number: MPC567xK_0N66N_0N72D
Rev. 31MAR2016