

## Mask Set Errata for Mask 1N36B

### Introduction

This report applies to mask 1N36B for these products:

- KINETIS72MHZ

Errata ID	Errata Title
3863	ADC: In 16-bit differential mode, ADC may result in a conversion error when positive input is near upper rail reference voltage
4588	DMAMUX: When using PIT with "always enabled" request, DMA request does not deassert correctly
5751	FTFx: Launching the Read 1's Section command (RD1SEC) on an entire flash block results in access error (ACCER).
4710	FTM: FTMx_PWMLOAD register does not support 8-/16-bit accesses
2793	I2C: MCU does not wake as expected from STOP or VLPS mode on subsequent address matches if previous address is mismatched
3964	JTAGC: When debug is active a wakeup from STOP or VLPS with interrupt causes a hard fault interrupt.
4590	MCG: Transitioning from VLPS to VLPR low power modes while in BLPI clock mode is not supported.
6665	Operating requirements: Limitation of the device operating range
5666	PMC: Maximum current consumption in VLPR, VLPW, VLPS, LLS and VLLS modes may be higher than data sheet specification.
4481	PMC: STOP mode recovery unstable
4638	PMC: VLLSx mode current draw at cold can exceed maximum specification at cold
5667	PMC: When used as an input to ADC or CMP modules, the PMC bandgap 1-V voltage reference is not available in VLPx, LLS, or VLLSx modes
4949	Reset and Boot: Device may not exit the power on reset (POR) event correctly with fast ramp-up slew rates.
5130	SAI: Under certain conditions, the CPU cannot reenter STOP mode via an asynchronous interrupt wakeup event
4218	SIM/FLEXBUS: SIM_SCGC7[FLEXBUS] bit should be cleared when the FlexBus is not being used.
6272	SIM: The System Integration Module Registers may not be programmed correctly.
5708	SLCD: LCD waveforms can exceed voltage specification for 3V or 5V glass

*Table continues on the next page...*

Errata ID	Errata Title
5952	SMC: Wakeup via the LLWU from LLS/VLLS to RUN to VLPR incorrectly triggers an immediate wakeup from the next low power mode entry
3926	TSI: The TSI will run several scan cycles during reference clock instead of scanning each electrode once
2638	TSI: The counter registers are not immediately updated after the EOSF bit is set.
4181	TSI: When the overrun flag is set, the TSI scanning sequence will exhibit undefined behavior.
4935	UART: CEA709.1 features not supported
2582	UART: Flow control timing issue can result in loss of characters
4647	UART: Flow control timing issue can result in loss of characters if FIFO is not enabled
4945	UART: ISO-7816 T=1 mode receive data format with a single stop bit is not supported
3892	UART: ISO-7816 automatic initial character detect feature not working correctly
5704	UART: TC bit in UARTx_S1 register is set before the last character is sent out in ISO7816 T=0 mode
5928	USBOTG: USBx_USBTRC0[USBRESET] bit does not operate as expected in all cases

### **e3863: ADC: In 16-bit differential mode, ADC may result in a conversion error when positive input is near upper rail reference voltage**

**Errata type:** Errata

**Description:** In 16-bit differential mode, the ADC may result in a conversion error when the input voltage on the plus-side of the differential pair (DPx) exceeds approximately ( $V_{REFH} \times 31/32$ ). Other modes are unaffected.

**Workaround:** To avoid a conversion error near positive full-scale in this mode, do not allow the input voltage on the plus-side of the differential pair (DPx) to exceed ( $V_{REFH} \times 31/32$ ).

### **e4588: DMAMUX: When using PIT with "always enabled" request, DMA request does not deassert correctly**

**Errata type:** Errata

**Description:** The PIT module is not assigned as a stand-alone DMA request source in the DMA request mux. Instead, the PIT is used as the trigger for the DMAMUX periodic trigger mode. If you want to use one of the PIT channels for periodic DMA requests, you would use the periodic trigger mode in conjunction with one of the "always enabled" DMA requests. However, the DMA request does not assert correctly in this case.

Instead of sending a single DMA request every time the PIT expires, the first time the PIT triggers a DMA transfer the "always enabled" source will not negate its request. This results in the DMA request remaining asserted continuously after the first trigger.

**Workaround:** Use of the PIT to trigger DMA channels where the major loop count is greater than one is not recommended. For periodic triggering of DMA requests with major loop counts greater than one, we recommended using another timer module instead of the PIT.

If using the PIT to trigger a DMA channel where the major loop count is set to one, then in order to get the desired periodic triggering, the DMA must do the following in the interrupt service routine for the DMA\_DONE interrupt:

1. Set the DMA\_TCDn\_CSR[DREQ] bit and configure DMAMUX\_CHCFGn[ENBL] = 0

2. Then again `DMAMUX_CHCFGn[ENBL] = 1`, `DMASREQ=channel` in your DMA DONE interrupt service routine so that "always enabled" source could negate its request then DMA request could be negated.

This will allow the desired periodic triggering to function as expected.

### **e5751: FTFx: Launching the Read 1's Section command (RD1SEC) on an entire flash block results in access error (ACCER).**

**Errata type:** Errata

**Description:** FTFx: Launching the Read 1's Section command on an entire flash block (i.e. with flash address = flash block base address & number of longwords = total number of longwords in the flash block) results in an incorrectly asserted access error (ACCER).

**Workaround:** To verify an entire flash block, use the Read 1's Block command. Use the Read 1's Section command only to verify sections that are smaller than an entire flash block.

### **e4710: FTM: FTMx\_PWMLOAD register does not support 8-/16-bit accesses**

**Errata type:** Errata

**Description:** The FTM PWM Load register should support 8-bit and 16-bit accesses. However, the `FTMx_PWMLOAD[LDOK]` bit is cleared automatically by FTM with these sized accesses, thus disabling the loading of the `FTMx_MOD`, `FTMx_CNTIN`, and `FTMx_CnV` registers.

**Workaround:** Always use a 32-bit write access to modify contents of the `FTMx_PWMLOAD` register.

### **e2793: I2C: MCU does not wake as expected from STOP or VLPS mode on subsequent address matches if previous address is mismatched**

**Errata type:** Errata

**Description:** The I2C module, acting as a slave on the I2C bus, does not wake as expected from normal STOP mode or VLPS mode on a valid address match if the previous address was not a match.

When the external I2C master sends a non-matching address, the I2C slave state machine does not look for a start bit past the first start bit on the bus. Consequently, subsequent transmissions by the I2C master with a matching address do not, on the first matching address, wake the MCU from stop mode or VLPS via the I2C interrupt.

**Workaround:** There are multiple workarounds:

(1) The master must continually re-transmit the MCU's slave address upon not receiving a NACK from the slave device during the slave addressing phase of the transmission. For clarification, the master must perform the following:

a) Send slave device address

b) Check for ACK bit

c) If ACK was received, continue with data transmission. Else, send repeated start signal and repeat steps a-c.

NOTE: Due to the nature of the errata, the maximum number of retransmissions needed to wake the part is nine times.

(2) When the MCU, operating as an I2C slave, is in STOP or VLPS mode: Ensure that the external I2C master sends a matching address to wake the slave MCU before it sends any transaction to other I2C slaves. The user must also ensure that MCU does not return to STOP or VLPS until after all packets to non-matching addresses have been sent.

(3) Use a pin interrupt (any pin, whether that pin is or is not being used by the active I2C module) to wake up the part before receiving I2C packets. NOTE: If using the SDA or SCL pin that the active I2C module is using, the part will wake-up on every I2C transaction on the bus.

(4) Use Wait mode instead of STOP or VLPS mode.

### **e3964: JTAGC: When debug is active a wakeup from STOP or VLPS with interrupt causes a hard fault interrupt.**

**Errata type:** Errata

**Description:** When exiting STOP or VLPS back into RUN mode with an interrupt a hard fault interrupt is caused when the JTAG debugger is enabled.

The MCU enters a pseudo STOP mode when the debugger is enabled.

The user cannot use the debugger to test code that wakes up from STOP with an interrupt.

**Workaround:** a. Disable the debugger with a Power off and on cycle before testing code that exits STOP or VLPS with an interrupt.

or

b. From the debugger, while in STOP or VLPS, halt the MCU with the debugger tools before triggering an interrupt.

### **e4590: MCG: Transitioning from VLPS to VLPR low power modes while in BLPI clock mode is not supported.**

**Errata type:** Errata

**Description:** Transitioning from VLPS mode back to VLPR (LPWUI control bit = 0) while using BLPI clock mode only, is not supported. During Fast IRC startup, the output clock frequency may exceed the maximum VLPR operating frequency. This does not apply to the BLPE clock mode.

**Workaround:** There are two options for workarounds

a) Exit to Run instead of VLPR. Before entering VLPR set the LPWUI bit so that when exiting VLPS mode the MCU exits to RUN mode instead of VLPR mode. With LPWUI set any interrupt will exit VLPR or VLPS back into RUN mode. To minimize the impact of the higher RUN current re-enter VLPR quickly.

or

b) Utilize MCG clock mode BLPE when transitioning from VLPS to VLPR modes.

### **e6665: Operating requirements: Limitation of the device operating range**

**Errata type:** Errata

**Description:** Some devices, when power is applied, may not consistently begin to execute code under certain voltage and temperature conditions. Applications that power up with either VDD  $\geq$  2.0 V or temperature  $\geq$  -20C are not impacted. Entry and exit of low-power modes is not impacted.

**Workaround:** To avoid this unwanted behavior, one or both of these conditions must be met:

- a) Perform power on reset of the device with a supply voltage (VDD) equal-to or greater-than 2.0 V , or
- b) Perform power on reset of the device at a temperature at or above -20 C.

#### **e5666: PMC: Maximum current consumption in VLPR, VLPW, VLPS, LLS and VLLS modes may be higher than data sheet specification.**

**Errata type:** Errata

**Description:** Maximum current consumption in Very Low Power Run (VLPR), Very Low Power Wait (VLPW), Very Low Power Stop (VLPS), Low Leakage Stop (LLS) ,Very Low Leakage Stop3 (VLLS3) ,Very Low Leakage Stop2 (VLLS2) ,Very Low Leakage Stop1 (VLLS1) , and Very Low Leakage Stop0 (VLLS0) modes within an operating range of -40°C to 25°C may exceed data sheet specification.

Note: Some devices do not feature all of the power modes listed above. Refer to the Reference Manual to determine if a particular low power mode is available on your device.

**Workaround:** If maximum current consumption in these low power modes exceed system requirements, a higher power mode should be used.

#### **e4481: PMC: STOP mode recovery unstable**

**Errata type:** Errata

**Description:** Recovery from STOP mode is not guaranteed if STOP mode is used for a period of time longer than 50ms.

**Workaround:** There are two methods that can be used:

1. Set the BGEN bit in the PMC\_REGSC register prior to entering STOP mode, and when existing STOP mode clear the BGEN bit.
2. Use a different low power mode such as VLPS.

#### **e4638: PMC: VLLSx mode current draw at cold can exceed maximum specification at cold**

**Errata type:** Errata

**Description:** When operating below -20C and above 3.4V in a VLLSx mode, current consumption can behave non-linearly and exceed maximum specification at cold. Current draw may approach maximum specification at hot.

**Workaround:** Limit operating temperature to -20C or greater or keep operating voltage below 3.4V, otherwise design power supply scheme should account for maximum specification as specified for hot.

**e5667: PMC: When used as an input to ADC or CMP modules, the PMC bandgap 1-V voltage reference is not available in VLPx, LLS, or VLLSx modes**

**Errata type:** Errata

**Description:** The Power Management Controller (PMC) bandgap 1-V reference is not available as an input to the Analog-to-Digital Converter (ADC) module (using ADC input channel AD27) or the Comparator (CMP) module (using CMP input IN6) in Very Low Power Run (VLPR), Very Low Power Wait (VLPW), Very Low Power Stop (VLPS), Low Leakage Stop (LLS), Very Low Leakage Stop3 (VLLS3), Very Low Leakage Stop2 (VLLS2), Very Low Leakage Stop1 (VLLS1), or Very Low Leakage Stop0 (VLLS0) modes.

This erratum does not apply to the VREF module 1.2 V reference voltage.

**Workaround:** Use of the PMC bandgap 1-V reference voltage as an input to the ADC and CMP modules requires the MCU to be in Run, Wait, or Stop modes.

**e4949: Reset and Boot: Device may not exit the power on reset (POR) event correctly with fast ramp-up slew rates.**

**Errata type:** Errata

**Description:** Device may not exit the power on reset (POR) event correctly when the Vdd ramp-up slew rate is greater than 17 kV/sec as VDD is raised from 0V to 1.7V.

**Workaround:** Keep instantaneous slew rate of VDD below 17 kV/sec.

Status: This errata will be fixed on future mask sets.

**e5130: SAI: Under certain conditions, the CPU cannot reenter STOP mode via an asynchronous interrupt wakeup event**

**Errata type:** Errata

**Description:** If the SAI generates an asynchronous interrupt to wake the core and it attempts to reenter STOP mode, then under certain conditions the STOP mode entry is blocked and the asynchronous interrupt will remain set.

This issue applies to interrupt wakeups due to the FIFO request flags or FIFO warning flags and then only if the time between the STOP mode exit and subsequent STOP mode reentry is less than 3 asynchronous bit clock cycles.

**Workaround:** Ensure that at least 3 bit clock cycles elapse following an asynchronous interrupt wakeup event, before STOP mode is reentered.

**e4218: SIM/FLEXBUS: SIM\_SCGC7[FLEXBUS] bit should be cleared when the FlexBus is not being used.**

**Errata type:** Errata

**Description:** The SIM\_SCGC7[FLEXBUS] bit is set by default. This means that the FlexBus will be enabled and come up in global chip select mode.

With some code sequence and register value combinations the core could attempt to prefetch from the FlexBus even though it might not actually use the value it prefetched. In the case where the FlexBus is unconfigured, this can result in a hung bus cycle on the FlexBus.

**Workaround:** If the FlexBus is not being used, disabled the clock to the FlexBus during chip initialization by clearing the SIM\_SCGC7[FLEXBUS] bit.

If the FlexBus will be used, then enable at least one chip select as early in the chip initialization process as possible.

## **e6272: SIM: The System Integration Module Registers may not be programmed correctly.**

**Errata type:** Errata

**Description:** The following System Integration Modules Registers may not be programmed correctly: the FAMID and PINID bit fields in the SIM\_SDID Register, the RAMSIZE in the SIM\_SOPT1 Register, the NVMSIZE, PFSIZE and EESIZE in SIM\_FCFG1 Register and information in the Unique ID Registers SIM\_UIDx.

**Workaround:** In next revision of silicon, these values will be programmed correctly.

## **e5708: SLCD: LCD waveforms can exceed voltage specification for 3V or 5V glass**

**Errata type:** Errata

**Description:** The LCD controller power supply can be sourced using VDD, internal regulated voltage VIREG, or an external supply on the VLL3. Using the internal Vireg (VSUPPLY[1:0] = 11, LCD\_GCR [RVEN] = 1, and LCD\_GCR[CPSEL] = 1) to generate the LCD bias voltages can cause the generated LCD waveform voltages to go out of specification for 3V and 5V glass. For LCD\_GCR[HREFSEL] = 0, Vireg = 1V, but bias voltages can exceed specification VLL1 > 1V, VLL2 > 2V, and VLL3 > 3V. For LCD\_GCR[HREFSEL] = 1 Vireg = 1.67V, but the bias voltages can exceed specification VLL1 > 1.67, VLL2 > 3.3V, and VLL3 > 5V.

**Workaround:** Use an alternate LCD power supply configuration.

1. Vdd can be used as the LCD controller power supply.
  - a. For VSUPPLY[1:0] = 00 and LCD\_GCR[CPSEL1] = 1, Vdd is used to drive VLL2. For 3V glass Vdd must be 2V and for 5V glass Vdd must be 3.3V.
  - b. For VSUPPLY[1:0] = 01 and LCD\_GCR[CPSEL1] = 1, Vdd is used to drive VLL3. For 3V glass Vdd must be 3V and for 5V glass Vdd must be 5V.
2. Drive VLL3 with an external supply VSUPPLY[1:0] = 11 and LCD\_GCR [RVEN] = 0.
  - a. For 3V glass VLL3 must equal 3V. The resistor bias network or charge pump can be used to generate VLL1 and VLL2. Note, VLL3 should never be externally driven to any level other than VDD.

## **e5952: SMC: Wakeup via the LLWU from LLS/VLLS to RUN to VLPR incorrectly triggers an immediate wakeup from the next low power mode entry**

**Errata type:** Errata



**Description:** Entering VLPR immediately after an LLWU wakeup event from LLS/VLLS, will cause any subsequent entry into LLS/VLLS to fail if entry into VLPR mode occurs before clearing the pending LLWU interrupt.

**Workaround:** After an LLWU wakeup event from LLS/VLLS, the user must clear the LLWU interrupt prior to entering VLPR mode.

## **e3926: TSI: The TSI will run several scan cycles during reference clock instead of scanning each electrode once**

**Errata type:** Errata

**Description:** The TSI will run several scan cycles during reference clock instead of scanning each electrode once. For each automatic scanning period determined by AMCLKS (clock source), AMPSC (prescaler) and SMOD (period modulo), TSI will scan during one reference clock cycle divided by the AMPSC prescaler.

This does not affect the count result from TSI because TSI counters keep the last scan result.

**Workaround:** 1. Because counter results are not affected, a simple workaround is to use the smallest prescaler possible and use a bigger SMOD value, this will minimize the number of extra scans, thus also minimizing the amount of average extra current used by the module.

2. If strict control of number of scan cycles is needed, trigger scans with software control (using the SWTS bit) and control time between scans with a separate timer. This solution is only recommended if strict control of scan cycles is needed, if not, recommendation is to use workaround 1.

## **e2638: TSI: The counter registers are not immediately updated after the EOSF bit is set.**

**Errata type:** Errata

**Description:** The counter registers are not immediately updated after the end of scan event (EOSF is set). The counter registers will become available 0.25 ms after the EOSF flag is set. This also applies for the end-of-scan interrupt, as it is triggered with the EOSF flag. This behavior will occur both in continuous scan and in software triggered scan modes.

**Workaround:** Insert a delay of 0.25 ms or greater prior to accessing the counter registers after an end of scan event or an end of scan interrupt that is triggered by the EOSF flag. This delay does not need to be a blocking delay, so it can be executing other actions before reading the counter registers. Notice that the out-of-range flag (OUTRGF) and interrupt occur after the counters have been updated, so if the OUTRGF flag is polled or the out-of-range interrupt is used, the workaround is not necessary.

## **e4181: TSI: When the overrun flag is set, the TSI scanning sequence will exhibit undefined behavior.**

**Errata type:** Errata

**Description:** When the overrun flag is set, the TSI scanning sequence will exhibit undefined behavior, so the results of measurements are invalid at this point. In order to continue reading valid measurements, disable the TSI module and reconfigure it.



**Workaround:** During development make sure to measure the required scanning time for all the electrodes in your system and configure the scanning time with AMCLKS, AMPSC and SMOD so that an overrun will not happen. Consider adding about 30 to 70% more time as headroom to make sure overrun is not triggered. If scanning time is critical and added scan time is not acceptable, detect the overrun condition either by polling the overrun flag in a loop or through the TSI interrupt. Once overrun is detected, disable the TSI module, clear all flags and reconfigure. During reconfiguration, SMOD can be increased by 10% or more of the current value to reduce the number of overrun occurrences.

### **e4935: UART: CEA709.1 features not supported**

**Errata type:** Errata

**Description:** Due to some issues that affect compliance with the specification, the CEA709.1 features of the UART module are not supported. Normal UART mode, IrDA, and ISO-7816 are unaffected.

**Workaround:** Do not use the UART in CEA709.1 mode.

### **e2582: UART: Flow control timing issue can result in loss of characters**

**Errata type:** Errata

**Description:** When /RTS flow control signal is used in receiver request-to-send mode, the /RTS signal is negated if the number of characters in the Receive FIFO is equal to or greater than the receive watermark. The /RTS signal will not negate until after the last character (the one that makes the condition for /RTS negation true) is completely received and recognized. This creates a delay between the end of the STOP bit and the negation of the /RTS signal. In some cases this delay can be long enough that a transmitter will start transmission of another character before it has a chance to recognize the negation of the /RTS signal (the /CTS input to the transmitter).

**Workaround:** For UARTs that implement an eight entry FIFO: When the FIFO is enabled, the receive watermark should be set to seven or less. This will ensure that there is space for at least one more character in the FIFO when /RTS negates. So in this case no data would be lost.

For UARTs without a FIFO (or if the FIFO is disabled): Delay might need to be added between characters on the transmit side in order to allow time for the negation of /RTS to be recognized before the next character is sent.

### **e4647: UART: Flow control timing issue can result in loss of characters if FIFO is not enabled**

**Errata type:** Errata

**Description:** On UART0 and UART1 when /RTS flow control signal is used in receiver request-to-send mode, the /RTS signal is negated if the number of characters in the Receive FIFO is equal to or greater than the receive watermark. The /RTS signal will not negate until after the last character (the one that makes the condition for /RTS negation true) is completely received and recognized. This creates a delay between the end of the STOP bit and the negation of the /RTS signal. In some cases this delay can be long enough that a transmitter will start transmission of another character before it has a chance to recognize the negation of the /RTS signal (the /CTS input to the transmitter).

**Workaround:** Always enable the Rx FIFO if you are using flow control for UART0 or UART1. The receive watermark should be set to seven or less. This will ensure that there is space for at least one more character in the FIFO when /RTS negates. So in this case no data would be lost.

Note that only UART0 and UART1 are affected. The UARTs that do not have the Rx FIFO feature are not affected.

#### **e4945: UART: ISO-7816 T=1 mode receive data format with a single stop bit is not supported**

**Errata type:** Errata

**Description:** Transmission of ISO-7816 data frames with single stop bit is supported in T=1 mode. Currently in order to receive a frame, two or more stop bits are required. This means that 11 ETU reception based on T=1 protocol is not supported. T=0 protocol is unaffected.

**Workaround:** Do not send T=1, 11 ETU frames to the UART in ISO-7816 mode. Use 12 ETU transmissions for T=1 protocol instead.

#### **e3892: UART: ISO-7816 automatic initial character detect feature not working correctly**

**Errata type:** Errata

**Description:** The ISO-7816 automatic initial character detection feature does not work. The direct convention initial character can be detected correctly, but the inverse convention initial character will only be detected if the S2[MSBF] and S2[RXINV] bits are set. This defeats the purpose of the initial character detection and automatic configuration of the S2[MSBF], S2[RXINV], and C3[TXINV] bits.

**Workaround:** Use software to manually detect initial characters. Configure the UART with S2[MSBF] and S2[RXINV] cleared. Then check UART receive characters looking for 0x3B or 0x03. If 0x3B is received, then the connected card is direct convention. If 0x03 is received, then the connected card is inverse convention. If an inverse convention card is detected, then software should set S2[MSBF], S2[RXINV], and C3[TXINV].

#### **e5704: UART: TC bit in UARTx\_S1 register is set before the last character is sent out in ISO7816 T=0 mode**

**Errata type:** Errata

**Description:** When using the UART in ISO-7816 mode, the UARTx\_S1[TC] flag sets after a NACK is received, but before guard time expires.

**Workaround:** If using the UART in ISO-7816 mode with T=0 and a guard time of 12 ETU, check the UARTn\_S1[TC] bit after each byte is transmitted. If a NACK is detected, then the transmitter should be reset.

The recommended code sequence is:

```
UART0_C2 &= ~UART_C2_TE_MASK; //make sure the transmitter is disabled at first
```

```
UART0_C3 |= UART_C3_TXDIR_MASK; //set the TX pin as output
```

```
UART0_C2 |= UART_C2_TE_MASK; //enable TX
```

```

UART0_C2 |= UART_C2_RE_MASK; //enable RX to detect NACK
for(i=0;i<length;i++)
{
while(!(UART0_S1&UART_S1_TDRE_MASK)){
UART0_D = data[i];
while(!(UART0_S1&UART_S1_TC_MASK)){//check for NACK
if(UART0_IS7816 & UART_IS7816_TXT_MASK)//check if TXT flag set
{
/* Disable transmit to clear the internal NACK detection counter */
UART0_C2 &= ~UART_C2_TE_MASK;
UART0_IS7816 = UART_IS7816_TXT_MASK;// write one to clear TXT
UART0_C2 |= UART_C2_TE_MASK; // re-enable transmit
}
}
UART0_C2 &= ~UART_C2_TE_MASK; //disable after transmit

```

## **e5928: USBOTG: USBx\_USBTRC0[USBRESET] bit does not operate as expected in all cases**

**Errata type:** Errata

**Description:** The USBx\_USBTRC0[USBRESET] bit is not properly synchronized. In some cases using the bit can cause the USB module to enter an undefined state.

**Workaround:** Do not use the USBx\_USBTRC0[USBRESET] bit. If USB registers need to be written to their reset states, then write those registers manually instead of using the module reset bit.

**How to Reach Us:****Home Page:**[freescale.com](http://freescale.com)**Web Support:**[freescale.com/support](http://freescale.com/support)

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

Freescale reserves the right to make changes without further notice to any products herein. Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale 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 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 does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: [freescale.com/SalesTermsandConditions](http://freescale.com/SalesTermsandConditions).

Freescale, the Freescale logo, Altivec, C-5, CodeTest, CodeWarrior, ColdFire, ColdFire+, C-Ware, Energy Efficient Solutions logo, Kinetis, mobileGT, PowerQUICC, Processor Expert, QorIQ, Qorivva, StarCore, Symphony, and VortiQa are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. Airfast, BeeKit, BeeStack, CoreNet, Flexis, Layerscape, MagniV, MXC, Platform in a Package, QorIQ Qonverge, QUICC Engine, Ready Play, SafeAssure, SafeAssure logo, SMARTMOS, Tower, TurboLink, Vybrid, and Xtrinsic are trademarks of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners.

© 2013 Freescale Semiconductor, Inc.