EFR32 Wireless Gecko EFR32MG21 Errata

This document contains information on the EFR32MG21 errata. The latest available revision of this device is revision B.

Errata that have been resolved remain documented and can be referenced for previous revisions of this device.

The device data sheet explains how to identify the chip revision, either from the package marking or electronically.

Errata effective date: November, 2022.

1. Errata Summary

The table below lists all known errata for the EFR32MG21 and all unresolved errata of the EFR32MG21.

Table 1.1. Errata Overview

Designator	Title/Problem	Workaround Exists	Exists on Revision:	
			A	B
CUR_E301	AVDD/IOVDD to DVDD Leakage Current	Yes	X	—
GPIO_E301	GPIO_PORTA_MODEL_MODE2 Write Affects SWDIO Pin During Active Debug	Yes	X	—
GPIO_E302	Increased Leakage Current When EM4WU Pins Are Enabled and the Pin State Is High	Yes	X	X
HFXO_E301	HFXO DISONDEMAND and FORCEEN Can Cause Device to Hang	Yes	X	X
I2C_E301	New Transfer Ignored if Bus Idle Timeout Occurs Between Start Detection and the Falling Edge of SCL	Yes	X	—
I2C_E302	Follower Holds SCL Low After Losing Arbitration	Yes	X	—
I2C_E303	I2C Fails to Indicate New Incoming Data	Yes	X	X
IADC_E301	Delta Sigma Modulator is Disabled in KEEPWARM Mode	Yes	X	—
IADC_E302	EM23ABORTERROR Interrupt Does Not Work	No	X	—
IADC_E303	Input Change Missed After Adjacent GND Conversions	No	X	—
IADC_E304	Possible Data Loss in EM2/EM3	Yes	X	X
IADC_E306	Changing Gain During a Scan Sequence Causes an Erroneous IADC Result	Yes	X	X
RADIO_E301	Improper TX and RX Operation at High Temperature	Yes	X	X
TIMER_E301	Continuous Overflow and Underflow Interrupts in Quadrature Counting Mode	Yes	X	X
USART_E301	Possible Data Transmission on Wrong Edge in Synchronous Mode	Yes	X	X
USART_E302	Additional SCLK Pulses Can Be Generated in USART Synchro-nous Mode	Yes	X	X
USART_E303	USART DMA Transactions Fail with Slow Peripheral Clocks	Yes	X	—
USART_E304	PRS Transmit Unavailable in Synchronous Secondary Mode	No	X	X
WDOG_E301	Clear Command is Lost Upon EM2 Entry	Yes	X	X

2. Current Errata Descriptions

2.1 GPIO_E302 – Increased Leakage Current When EM4WU Pins Are Enabled and the Pin State Is High

Description of Errata
When any of the EM4WU pins are used with the input path enabled and the pin state is high, an extra leakage current of approximately 15 µA per pin will be observed in EM0, EM1, EM2, and EM3.
Affected Conditions / Impacts
EM0, EM1, EM2, and EM3 current will be higher by approximately 15 µA per pin when any of the EM4WU pins are used with the input path enabled and the pin state is high.
Workaround
There are two workarounds for this issue: 1. If the input path on the pad is not required, disable the input path on that pad by setting the DINDIS or DINDISALT bits in the GPIO_PORTx_CTRL register. Thus, an EM4WU pin can still be used to drive an output without incurring the extra current leakage when the pin is configured as an output and DINDIS or DINDISALT is set. 2. If an input path is required (i.e., MODEn is any value other than DISABLED and DINDIS = 0 or DINDISALT = 0), assign it to a pin which does not have EM4 wakeup capability. Refer to the device data sheet to determine which pins have or do not have EM4 wake-up functionality.
Resolution
There is currently no resolution for this issue.

2.2 HFXO_E301 — HFXO DISONDEMAND and FORCEEN Can Cause Device to Hang

Description of Errata
With HFXO enabled, when DISONDEMAND is toggled from 0 to 1 followed by a system reset request, a handshake between the EMU and CMU hangs, preventing the system reset from being asserted.
Affected Conditions / Impacts
The device will hang waiting for the EMU/CMU handshake to complete, requiring a pin reset to recover.
Workaround
When the HFXO is enabled, do not toggle DISONDEMAND from 0 to 1.
Resolution
There is currently no resolution for this issue.

2.3 I2C_E303 – I^2C Fails to Indicate New Incoming Data

Description of Errata
A race condition exists in which the I^2C fails to indicate reception of new data when both user software attempts to read data from and the I^2C hardware attempts to write data to the I2C_RXFIFO in the same cycle.
Affected Conditions / Impacts
When this race condition occurs, the RXFIFO enters an invalid state in which both I2C_STATUS_RXDATAV = 0 and I2C_STATUS_RXFULL = 1. This causes the I^2C to discard new incoming data bytes because RXFULL = 1 and would otherwise prevent user software from reading last byte written by the I^2C hardware to RXFIFO because RXDATAV = 0.
Workaround
User software can recognize and clear this invalid RXDATAV = 0 and RXFULL = 1 condition by performing a dummy read of the RXFIFO (I2C_RXDATA). This restores the expected RXDATAV = 1 and RXFULL = 0 condition. The dummy read also sets the RXUFIF flag bit, which should be ignored and cleared. The data from this read can be discarded, and user software can now read the last byte written by the I^2C hardware to the RXFIFO (the byte which caused the invalid RXDATAV = 0 and RXFULL = 1 condition). No data will be lost as long as user software completes this recovery procedure (performing the dummy read and then reading the remaining valid byte in the RXFIFO) before the I^2C hardware receives the next incoming data byte.
Resolution
There is currently no resolution for this issue.

2.4 IADC_E304 – Possible Data Loss in EM2/EM3

Description of Errata
When the IADC wakes from EM2 or EM3 and generates conversion results that the LDMA transfers to RAM, it is possible under very rare circumstances to lose data when the ratio of the bus clock (HCLK) is slow compared to the prescaled IADC clock (ADC_CLK).
Affected Conditions / Impacts
Data from IADC conversions in these cases can potentially be lost due to FIFO overflow.
Workaround
To prevent data loss when the IADC awakens from EM2 or EM3 and performs conversions that are serviced by the LDMA before re-entering the low-energy state, make sure that: • the rate at which the IADC takes samples in EM2 or EM3 is less than or equal to 125 kHz (samples are taken no faster than every 8 µs), and • the frequency of the HCLK (bus clock) is at least four times the frequency of the IADCCLK.
Resolution
There is currently no resolution for this issue.

2.5 IADC_E306 – Changing Gain During a Scan Sequence Causes an Erroneous IADC Result

Description of Errata
Differences in the ANALOGGAIN setting within multiple IADC_CFGx groups during a scan sequence introduces a transient condition that may result in an inaccurate IADC conversion.
Affected Conditions / Impacts
The result of the IADC scan measurement may not match the expected result for the voltage present on the pin during the conversion.
Workaround
Both 1 and 2 shown below must be implemented. 1. If there is a difference in the ANALOGGAIN setting between IADC_CFGx groups during a scan sequence, the IADC_SCHEDx clock prescaler must also change to an appropriate setting. This forces a warmup state (5 µs delay) in between ANALOGGAIN changes. Note that the same IADC_SCHEDx clock prescaler value may be an appropriate setting for both ANALOGGAIN settings, but to force the warmup delay, the IADC_SCHEDx must have different values. 2. The first and last entry of a scan group should use IADC_CFG0, which is the default configuration of the IADC at the start and end of a scan conversion sequence. If CONFIG1 is used at the start and end of the scan group, erronous IADC results may occur.
Resolution
There is currently no resolution for this issue.

2.6 RADIO_E301 – Improper TX and RX Operation at High Temperature

Description of Errata
Some radio transceivers may fail to lock to the correct RF frequency at high operating temperatures when using Gecko SDK prior to Gecko SDK v2.5.4.
Affected Conditions / Impacts
Devices using Gecko SDK prior to v2.5.4 at high operating temperatures may be unable to lock to the desired RF frequency. This may cause errors in the TX/RX frequency or an inability to transmit or receive data.
Workaround
Customers should use firmware provided in Gecko SDK v2.5.4 or later for proper TX/RX operation.
Resolution
There is currently no resolution for this issue.

2.7 TIMER_E301 — Continuous Overflow and Underflow Interrupts in Quadrature Counting Mode

Description of Errata
When the TIMER is configured to operate in quadrature decoder mode with the overflow interrupt enabled and the counter value (TIMER_CNT) reaches the top value (TIMER_TOP), the overflow interrupt is requested contiunously even if the interrupt flag (TIMER_IF_OF) is cleared. Similarly, if the underflow interrupt is enabled and the counter value reaches zero, the underflow interrupt is requested contiunously even if the interrupt flag (TIMER_IF_UF) is cleared. Only after the counter value has incremented or decremented so that the overflow or underflow condition no longer applies can the interrupt be cleared.
Affected Conditions / Impacts
Because the counter is clocked by its CC0 and CC1 inputs in quadrature decoder mode and not the prescaled HFPERCLK, overflow and underflow events remain latched as long TIMER_CNT remains at the value that triggered the overflow or underflow condition. Until the counter is no longer in the overflow or underflow condition, it is not possible to clear the associated interrupt flag.
Workaround

Short of disabling the relevant interrupts, the simplest workaround is to manually increment or decrement TIMER_CNT so that the overflow or underflow condition no longer exists. Insert the following or similar code in the interrupt handler for the timer in question (TIMER0 in this case) to do this:

```
uint32 intflags = TIMER_IntGet(TIMER0);

if (intFlags & TIMER_IEN_OF)
  TIMER0->CNT += 1;

if (intFlags & TIMER_IEN_UF)
  TIMER0->CNT -= 1;
```

It may be necessary for firmware to account for this adjustment in calculations that include the counter value.

Resolution
There is currently no resolution for this issue.

2.8 USART_E301 — Possible Data Transmission on Wrong Edge in Synchronous Mode

Description of Errata
The first bit of the new data word is incorrectly transmitted on the leading clock edge of the subsequent data bit and not the trailing clock edge of the current data bit if the USART is configured to operate in synchronous mode with 1. USART_CLKDIV_DIV = 0 (clock = $f_{HFPERCLK}$ ÷ 2), 2. USART_CTRL_CLKPHA = 0, 3. USART_TIMING_CSHOLD = 1 and 4. Data is loaded into the transmit FIFO (say, by the LDMA) at the exact same time as the USART state machine begins to insert the requested one bit time extension of the chip select hold time (USART_TIMING_CSHOLD = 1).
Affected Conditions / Impacts
Reception of each data bit by the secondary is tied to a specific clock edge. Therefore, the late transmission by the main of the first bit of a word may cause the secondary to receive the incorrect data, especially if the data setup time for the secondary approaches or exceeds one half the shift clock period.
Workaround
Because there is no way to specifically time a write to the transmit FIFO such that it does not occur when the USART state machine changes state, use one of the following workarounds to avoid the risk for data corruption described above: • Set USART_CLK_DIV > 0. • Use USART_TIMING_CSHOLD = 0 or USART_TIMING_CSHOLD > 1. • Use USART_CTRL_CLKPHA = 1. This option is particularly useful with SPI flash memories as many support operation in both the CLKPOL = CLKPHA = 0 and CLKPOL = CLKPHA = 1 modes.
Resolution
There is currently no resolution for this issue.

2.9 USART_E302 — Additional SCLK Pulses Can Be Generated in USART Synchronous Mode

Description of Errata
When inter-character spacing is enabled (USART_TIMING_ICS > 0) and USART_CTRL_CLKPHA = 1 in synchronous main mode, an extra clock pulse is generated after each frame transmitted except the last (that frame which when sent results in both the transmit FIFO and transmit shift register being empty).
Affected Conditions / Impacts
The extra clock pulse generated at the end of the first frame would cause a secondary device to clock in the first bit of the next frame it expects to receive even though the USART is not yet driving that data. The secondary would lose synchronization with the main and erroneously receive all frames after the first.
Workaround
Do not enable inter-character spacing when CLKPHA = 1. If a delay between frames is necessary, insert one manually with a software delay loop. Data cannot be transmitted using DMA in this case.
Resolution
There is currently no resolution for this issue.

2.10 USART_E304 — PRS Transmit Unavailable in Synchronous Secondary Mode

Description of Errata
When the USART is configured for synchronous secondary operation, the transmit output (MISO) is not driven if the signal is routed to a pin using the PRS producer (e.g., SOURCESEL = 0x20 and SIGSEL = 0x4 for USART0).
Affected Conditions / Impacts
Systems cannot operate the USART in synchronous secondary mode if the PRS is used to route the transmit output to the RX (MISO) pin. Operation is not affected in main mode when the transmit output is routed to the TX (MOSI) pin using the PRS producer nor is operation affected in any mode when the GPIO_USARTn_RXROUTE and GPIO_USARTn_TXROUTE registers are used.
Workaround
There is currently no workaround for this issue.
Resolution
There is currently no resolution for this issue.

2.11 WDOG_E301 – Clear Command is Lost Upon EM2 Entry

Description of Errata
If the device enters EM2, while the clear command is still being synchronized, the watchdog counter may not be cleared as expected.
Affected Conditions / Impacts
If the watchdog counter is not cleared as expected, the device can encounter a watchdog reset.
Workaround
Wait for WDOG_SYNCBUSY_CMD to clear before entering EM2. Note that WDOG can be clocked from one of the low-frequency clock sources and will require additional time to enter EM2 when implementing this workaround.
Resolution
There is currently no resolution for this issue.

3. Resolved Errata Descriptions

This section contains previous errata for EFR32MG21 devices.

For errata on the latest revision, refer to the beginning of this document. The device data sheet explains how to identify chip revision, either from package marking or electronically.

3.1 CUR_E301 – AVDD/IOVDD to DVDD Leakage Current

Description of Errata
Leakage from AVDD or IOVDD to DVDD is present when either supply voltage is higher than DVDD.
Affected Conditions / Impacts
When the AVDD or IOVDD supply voltage is higher than DVDD, a leakage current from AVDD or IOVDD to DVDD is present. This current has a diode-like property, such that when the voltage difference is less than 700 mV, the leakage is less than 1 µA. If the difference is near the maximum (e.g. AVDD = 3.8 V and DVDD = 1.8 V), the leakage can be as high as 100 µA on a typical device at room temperature. In this case, there is also as much as 50 µA of added current from AVDD or IOVDD directly to ground.
Workaround
Enable the AVDD and/or IOVDD brownout detector via the EMU_BOD3SENSE register for the supply voltage(s) that is/are higher than DVDD. • Enable the AVDD monitor by performing a read-modify-write operation on the EMU_BOD3SENSE with 0x1 as the bit mask. • Enable the IOVDD monitor by performing a read-modify-write operation on the EMU_BOD3SENSE with 0x6 as the bit mask. Note that enabling the relevant brownout detector minimizes this leakage current, but it does not eliminate it completely.
Resolution
This issue is resolved in revision B devices.

3.2 GPIO_E301 – GPIO_PORTA_MODEL_MODE2 Write Affects SWDIO Pin During Active Debug

Description of Errata
When a debugger is connected to the device, software cannot clear GPIO_DBGROUTEPEN in order to prevent loss of communication with the host. However, changing the GPIO_PORTA_MODEL_MODE2 field, which corresponds to the SWDIO pin, to any of the wired-AND/wired-OR modes effectively disables the debugger connection.
Affected Conditions / Impacts
Reconfiguring the SWDIO pin to a wired-AND/wired-OR output mode causes loss of debugger contact upon writing to the GPIO_PORTA_MODEL_MODE2 field.
Workaround
To prevent the debugger from losing its connection to the target device, do not change the state of the GPIO_PORTA_MODEL_MODE2 field.
Resolution
This issue is resolved in revision B devices.

3.3 I2C_E301 – New Transfer Ignored if Bus Idle Timeout Occurs Between Start Detection and the Falling Edge of SCL

Description of Errata
If a bus idle timeout occurs between detection of a start condition and the falling edge of SCL, the start condition detection logic is defeated, causing the I^2C state machine to indicate that bus is not busy (I2C_STATE_BUSY = 0).
Affected Conditions / Impacts
A transfer that meets the timing conditions cited above will be missed, causing the device not to respond to the leader if it is the follower being addressed. Furthermore, because I2C_STATE_BUSY no longer reflects the actual state of the bus, the device can, if configured as a leader, mistakenly attempt to use the bus, thus corrupting a transfer already in progress.
Workaround
To avoid corrupting bus activity, application software should implement the following before starting a transaction in systems where the bus timeout is used: • Wait for the I2C_IF_SSTOP flag, either by polling or by using the associated interrupt (I2C_IEN_SSTOP). • Impose a system-defined delay after all transfers that are independent of the bus timeout monitor to ensure that the bus is in idle state. When one of the above workarounds is met, the bus can be considered inactive and available for use.
Resolution
This issue is resolved in revision B devices.

3.4 I2C_E302 – Follower Holds SCL Low After Losing Arbitration

Description of Errata
If, while transmitting data as a follower, arbitration is lost, SCL is unintentionally held low for an indefinite period of time.
Affected Conditions / Impacts
The winner of arbitration cannot use the bus because SCL is never released.
Workaround
If the I^2C arbitration lost flag is asserted (I2C_IF_ARBLOST = 1) in follower mode (I2C_STATE_MASTER = 0), application software needs to wait for at least one SCL high time and then issue the transmission abort command (set I2C_CMD_ABORT = 1), thus releasing SCL.
Resolution
This issue is resolved in revision B devices.

3.5 IADC_E301 – Delta Sigma Modulator is Disabled in KEEPWARM Mode

Description of Errata
When IADC_CTRL_WARMUPMODE = KEEPWARM, the IADC delta sigma modulator is disabled between conversions.
Affected Conditions / Impacts
Because the delta sigma modulator is disabled before conversions restart, the results will be erroneous until the usual 1 μs required for warm-up has elapsed.
Workaround
Do not use IADC_CTRL_WARMUPMODE = KEEPWARM or discard the results received during the first 1 μs of operation after restarting the converter.
Resolution
This issue is resolved in revision B devices.

3.6 IADC_E302 – EM23ABORTERROR Interrupt Does Not Work

Description of Errata
When IADC_IEN_EM23ABORTERROR = 1, the IADC does not request an interrupt upon EM2 or EM3 entry when running from a clock that is not active in these energy modes.
Affected Conditions / Impacts
There is no way for the IADC to let application software know that the system has (erroneously) entered EM2 or EM3 with a converter clock source that is now disabled.
Workaround
There is currently no workaround for this issue.
Resolution
This issue is resolved in revision B devices.

3.7 IADC_E303 – Input Change Missed After Adjacent GND Conversions

Description of Errata
If the IADC is performing a scan that includes two adjacent GND conversions (IADC_SCAN[n]_PORTPOS = IADC_SCAN[n]_PORTNEG = GND and IADC_SCAN[n + 1]_PORTPOS = IADC_SCAN[n + 1]_PORTNEG = GND) such that the configuration for one GND conversion differs from the other (e.g. IADC_SCAN[n]_CFG = 0 and IADC_SCAN[n + 1]_CFG = 1 or vice versa), the inputs for conversion [n + 2] in the sequence after the two GND conversions will not be selected.
Affected Conditions / Impacts
Results for the first conversion after two adjacent GND conversions in a scan will be erroneous.
Workaround
Do not perform scans that include two adjacent GND conversions.
Resolution
This issue is resolved in revision B devices.

3.8 USART_E303 — USART DMA Transactions Fail with Slow Peripheral Clocks

Description of Errata
USART DMA transactions will fail when the USART peripheral clock is slower than the DMA clock and IGNORESREQ is cleared to 0.
Affected Conditions / Impacts
Systems will not be able to use the DMA with a USART running from a slow clock when IGNORESREQ is cleared to 0.
Workaround
Use one of the following options to avoid USART DMA transaction failures: • Set IGNORESREQ to 1 in LDMA. • Do not prescale USART clock.
Resolution
This issue is resolved in revision B devices.

4. Revision History

Revision 0.8

November, 2022

- Updated errata description and workaround for HFXO_E301.
- Updated workaround for IADC_E306.

Revision 0.7

March, 2022

- Updated the workaround in I2C_E303.
- Added USART_E304.
- Added IADC_E306.
- Replaced select terms with inclusive lexicon.

Revision 0.6

August, 2020

- Added I2C_E303.
- Clarified the affected conditions and impacts in WDOG_E301.

Revision 0.5

June, 2020

- Added TIMER_E301, USART_E301, USART_E302, USART_E303 and WDOG_E301.
- Migrated to new errata document format.

Revision 0.4

May, 2019

- Added RADIO_E301.

Revision 0.3

April, 2019

- Added GPIO_E302 and HFXO_E301.

Revision 0.2

December, 2018

- Updated for device revision B.
- Added IADC_E303 and IADC_E304.
- CUR_E301, GPIO_E301, I2C_E301, I2C_E302, IADC_E301, IADC_E302, and IADC_E303 resolved and moved to 3. Resolved Errata Descriptions.

Revision 0.1

May, 2018

- Initial release.

Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support & Community
www.silabs.com/community

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

www.silabs.com