SILICON LABS

EFM32G Errata

This document contains information on the EFM32G errata. The latest available revision of this device is revision E.

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 package marking or electronically.

Errata effective date: March 2021.

1. Errata Summary

The table below lists all known errata for the EFM32G and all unresolved errata in revision E of the EFM32G.

Table 1.1. Errata Overview

Designator	Title/Problem	Workaround Exists	Exists on Revision:				
			A	B	C	D	E
ACMP_E101	ACMP Mode	Yes	—	X	—	—	—
ADC_E101	ADC Temperature Sensor	No	X	—	—	—	—
ADC_E102	ADC SCANGAIN	Yes	X	—	—	—	—
ADC_E104	ADC 1 Msample/s	Yes	X	X	—	—	—
ADC_E105	ADC Output	Yes	X	—	—	—	—
ADC_E106	ADC Reference Settling	Yes	X	X	—	—	—
ADC_E108	ADC Temperature Sensor	Yes	—	X	—	—	—
ADC_E110	ADC VDD Reference Gives Half Resolution	Yes	—	X	—	—	—
ADC_E111	ADC References Doubled	Yes	X	X	—	—	—
ADC_E112	ADC Accuracy	Yes	X	X	—	—	—
ADC_E113	ADC Variability	Yes	X	X	—	—	—
ADC_E114	Incorrect ADC Calibration Register Reset Value	Yes	X	X	—	—	—
ADC_E115	Incorrect ADC Temperature Sensor Calibration Data	Yes	X	X	X	—	—
ADC_E118	Requirements for ADC_CLK > 7 MHz	Yes	X	X	X	X	X
AES_E101	BYTEORDER Does Not Work in Combination with DATASTART/XORSTART	Yes	X	X	X	X	X
BOD_E101	BOD Threshold	Yes	X	—	—	—	—
CMU_E101	Peripheral Clocks Active In EM2/EM3 During Debug	Yes	X	—	—	—	—
CMU_E102	LFRCO/HFRCO Frequency Change During EM2/3	Yes	X	—	—	—	—
CMU_E103	Wrong RCO Frequency	Yes	X	—	—	—	—
CMU_E104	Energy Mode Transitions Cause HFRCO Overshoot	Yes	X	X	—	—	—
CMU_E105	AUXHFRCO Active in EM2/EM3	Yes	X	—	—	—	—
CMU_E106	LFXO Digital External Mode	Yes	X	X	X	X	X
CMU_E107	Disabled Low Frequency Clocks	Yes	X	X	—	—	—
CMU_E108	LFxCLKEN Write	Yes	X	X	X	—	—
CMU_E109	LFXO Configuration Incorrect	Yes	X	X	X	—	—
CMU_E115	HFRCO 1 MHz Band Switching	Yes	X	X	X	X	X
DAC_E101	DAC Sample-Hold	Yes	X	X	—	—	—
DAC_E102	DAC Enabling	Yes	X	X	—	—	—
DAC_E103	DAC Ringing	Yes	X	X	—	—	—

Designator	Title/Problem	Work-around Exists	Exists on Revision:				
			A	B	C	D	E
DAC_E104	DAC Sample-Hold/Sample-Off	Yes	X	X	—	—	—
DAC_E105	DAC Startup	Yes	X	X	—	—	—
DAC_E107	DAC Accuracy	No	X	X	—	—	—
DAC_E108	Incorrect DAC Calibration Register Reset Value	Yes	X	X	—	—	—
DAC_E109	DAC Output Drift Over Lifetime	Yes	X	X	X	X	X
DMA_E101	EM2 with WFE and DMA	Yes	X	X	X	X	X
EBI_E101	EBI Lines	Yes	X	X	—	—	—
EMU_E101	EM Transition Brown Out	Yes	X	X	—	—	—
EMU_E102	DMA Clock EM2/EM3	Yes	X	—	—	—	—
EMU_E103	EM4 current	Yes	—	X	X	X	X
EMU_E104	Sequencing of Analog and Digital Power	Yes	X	X	X	X	X
EMU_E105	Debug Unavailable During DMA Processing from EM2	Yes	X	X	X	—	—
EMU_E106	SWO Line Pulled Low in EM2	Yes	X	X	X	—	—
EMU_E107	Interrupts During EM2 Entry	Yes	X	X	X	X	—
HFRCO_E101	HFRCO Calibration	Yes	X	—	—	—	—
I2C_E101	I2C RX Overflow	Yes	X	—	—	—	—
I2C_E102	I2C Disabled After EM2/EM3	Yes	X	X	—	—	—
LCD_E101	LCD Com Line	Yes	X	—	—	—	—
LCD_E102	LCD Voltage Boost Current	Yes	X	X	—	—	—
LCD_E103	Indeterminate Animation Engine Start-Up	Yes	X	X	X	X	X
LCD_E104	Increased Current Draw when VLCD > VDDIO and LCD Pins are Used for GPIO	Yes	X	X	X	X	X
LEUART_E101	LEUART + DMA	No	X	—	—	—	—
LEUART_E102	LEUART Baudrate	Yes	X	X	—	—	—
LEUART_E103	LEUART RXOF	Yes	X	X	—	—	—
LETIMER_E101	Buffered Top Value	Yes	X	X	—	—	—
LFRCO_E101	LFRCO Frequency	Yes	X	X	—	—	—
LFXO_E102	LFXO Temperature Sensitivity	Yes	X	X	—	—	—
PCNT_E101	PCNT0 TOP Register	Yes	X	X	—	—	—
PCNT_E102	PCNT Pulse Width Filtering Does Not Work	No	X	X	X	X	X
RTC_E101	RTC PRS Output	Yes	X	X	X	X	X
TIMER_E101	TIMER Up/Down Mode	Yes	X	X	—	—	—
TIMER_E102	Timer Capture and Debugger	Yes	X	X	X	X	X
USART_E101	U(S)ART Double Buffer	Yes	X	X	X	X	X
USART_E102	U(S)ART RXOF	Yes	X	X	—	—	—

Designator	Title/Problem	Work-around Exists	Exists on Revision:				
			A	B	C	D	E
USART_E103	U(S)ART Slave TXUF Causes Shift	Yes	X	X	—	—	—
USART_E104	U(S)ART AUTOTRI One Cycle Late	Yes	X	X	—	—	—
USART_E105	U(S)ART Fractional Baudrate	Yes	X	X	—	—	—
USART_E106	U(S)ART Slave TX Tristate	Yes	X	X	—	—	—
USART_E107	U(S)ART TXC	Yes	X	X	—	—	—
USART_E108	U(S)ART Slave TX Data Required Early	Yes	X	X	—	—	—
USART_E109	U(S)ART Slave TXUF Artifacts	Yes	X	X	—	—	—
USART_E110	U(S)ART Slave TX Data Lost	Yes	X	X	—	—	—
USART_E111	U(S)ART RX DMA Request After TX	Yes	X	X	—	—	—
VCMP_E101	VCMP Mode	Yes	—	X	—	—	—
VCMP_E102	VCMP Current	No	X	X	—	—	—
WDOG_E101	WDOG in EM2/EM3	Yes	X	—	—	—	—
WDOG_E102	WDOG Does Not Freeze in EM2/EM3	Yes	X	X	X	—	—
WDOG_E103	WDOG EM2 Detection with LFXO Digital/Sine Input	Yes	X	X	X	X	X

2. Current Errata Descriptions

2.1 ADC_E118 — Requirements for ADC_CLK > 7 MHz

Description of Errata
If operating the ADC_CLK at frequencies greater than 7 MHz, the ADC_BIASPROG register must be set to a value of 0xF4B in order to meet specified performance. Operating the ADC_CLK at frequencies of 7 MHz or lower can use the default ADC_BIASPROG value of 0x747.
Affected Conditions / Impacts
Devices operating the ADC_CLK at frequencies greater than 7 MHz while using the default ADC_BIASPROG value of 0x747 may experience performance outside data sheet limits.
Workaround
For systems requiring an ADC_CLK rate > 7 MHz, set the ADC_BIASPROG register to 0xF4B. For systems requiring an ADC_CLK rate ≤ 7 MHz, set the ADC_BIASPROG register to 0x747.
Resolution
The revision 2.0 datasheet has been updated to reflect the new recommendations.

2.2 AES_E101 — BYTEORDER Does Not Work in Combination with DATASTART/XORSTART

Description of Errata
When the BYTEORDER bit in AES_CTRL is set, an encryption or decryption should not be started through DATASTART or XOR-START.
Affected Conditions / Impacts
If BYTEORDER is used in combination with DATASTART or XORSTART, the AES data and key are interpreted in the wrong order.
Workaround
Do not use BYTEORDER in combination with DATASTART or XORSTART.
Resolution
There is currently no resolution for this issue.

2.3 CMU_E106 — LFXO Digital External Mode

Description of Errata
LFXO ready flags are never set when LFXO is configured in Digital External Clock mode.
Affected Conditions / Impacts
When LFXOMODE in CMU_CTRL is set to DIGEXTCLK, the LFXORDY flag in CMU_STATUS and CMU_IF will not be set when the number of cycles set in LFXOTIMEOUT in CMU_CTRL has elapsed. Thus, polling of this flag will not work. However, the clock propagates as normal. It is only the flag that is not set.
Workaround
To detect that the clock has propagated through the ripple counter, write to any Asynchronous Register in Low Energy peripheral and wait for SYNCBUSY for that register field to go low. Remember to enable the LE core clockand the clock for the LE peripheral you choose. For example, write 0xA5 to RTC_COMP0 and wait for COMP0 in RTC_SYNCBUSY to go low.
Resolution
There is currently no resolution for this issue.

2.4 CMU_E115 — HFRCO 1 MHz Band Switching

Description of Errata
Switching to or from the 1 MHz band of the HFRCO or AUXHFRCO may cause a hard fault even at the maximum supported number of wait states.
Affected Conditions / Impacts
When the HFRCO or AUXHFRCO is selected as a clock source (e.g., the HFRCO has been selected as the HFCLK source) and the device is running with the maximum supported number of wait states, switching to or from the 1 MHz band can possibly cause a clock glitch that results in unexpected behavior or a hard fault.
Workaround
Before switching to or from the 1 MHz band when the HFRCO or AUXHFRCO is selected as the clock source, first switch to another stable clock source (such as the LFRCO). For example, when switching from the 21 MHz band to the 1 MHz band, the following procedure needs to be followed: 1. Select another stable clock source by writing to the HFCLKSEL field of the CMU_CMD register. 2. Wait until the clock source shows that it has been selected in the CMU_STATUS register, (e.g., CMU_STATUS_LFRCOSEL = 1). 3. Program the CMU_HFRCOCTRL register to select the 1 MHz band and tuning value. 4. Wait until the HFRCO has stabilized at the new frequency by waiting for the HFRCORDY bit in the CMU_STATUS register to change for 0 to 1. 5. Select the HFRCO as the clock source by writing to the HFCLKSEL field of the CMU_CMD register.
Resolution
There is currently no resolution for this issue.

2.5 DAC_E109 — DAC Output Drift Over Lifetime

Description of Errata
The voltage output of the DAC might drift over time.
Affected Conditions / Impacts
When the device is powered and the DAC is disabled, stress on an internal circuit node can cause the output voltage of the DAC to drift over time, and in some cases may violate the $V_{DACOFFSET}$ specification. If the DAC is always enabled while the device is powered, this condition cannot occur.
Workaround
Both in the startup initialization code and prior to disabling the DAC in application code, set the OPAnSHORT bit in DACn_OPACTRL to a '1' for the corresponding DAC(s) used by the application. This will prevent the output voltage drift over time effect.
Resolution
There is currently no resolution for this issue.

2.6 DMA_E101 — EM2 with WFE and DMA

Description of Errata
WFE does not work for the DMA in EM2.
Affected Conditions / Impacts
In EM2, when sleeping with WFE (Wait for Event), an interrupt from the DMA will not wake up the system.
Workaround
Use WFI (Wait for Interrupt) or EM1 instead.
Resolution
There is currently no resolution for this issue.

2.7 EMU_E103 — EM4 current

Description of Errata
In EM4 the device may consume 700nA instead of 20nA.
Affected Conditions / Impacts
If EM4 is issued within a 10 µs - 12 µs window after the 1 kHz RC oscillator rising edge transition, the device will permanently consume 700 nA.
Workaround
There are two possible workarounds for this issue: 1. The first workaround is using the WDOG to identify the rising edge transition and add a delay before going into EM4. Write on the WDOG_CTRL register (for instance `WDOG->CTRL|=WDOG_CTRL_CLKSEL_ULFRCO`) and wait for the SYNCBUSY to be released. The release of the SYNCBUSY happens on a rising edge transition of the 1 kHz clock. After that, insert a number of `__NOP();` to cause a delay of 20 µs (12 µs plus margin). The number of `__NOP();` will depend on the processor frequency. After the delay, EM4 can be entered safely. **Note:** To implement this workaround, the WDOG cannot be locked; otherwise, the registers will not be written. 2. The second workaround option is to output the ULFRCO on a pin (CMU_CLK0) using CMU_CTRL and CMU_ROUTE registers. That pin should then be configured as push pull with interrupt enable on rising edge, so the device can go to EM2 while it waits for the ULFRCO rising edge transition. When the interrupt occurs, clear it and add a number of `__NOP();` before entering EM4, as described in the first workaround. **Note:** The pin used to output the ULFRCO should not be driven by an external source.
Resolution
There is currently no resolution for this issue.

2.8 EMU_E104 — Sequencing of Analog and Digital Power

Description of Errata
Power-on Reset might fail if power is applied to IOVDD_x or VDD_DREG before AVDD_x.
Affected Conditions / Impacts
The device might lock up if power is applied to IOVDD_x or VDD_DREG pins before AVDD_x pins during power up. This lock-up state can be exited by removing power to the device followed by a power up sequence according to what is described in the workaround.
Workaround
Make sure that the power on the AVDD_x pins ramp earlier or at the same time as the power on IOVDD_x and VDD_DREG during power up. Practical schematic recommendations for this workaround are given in the EFM32 Application Note *AN0002: Hardware Design Considerations*.
Resolution
There is currently no resolution for this issue.

2.9 LCD_E103 — Indeterminate Animation Engine Start-Up

Description of Errata
The LCD controller animation engine starts counting based on when the writes to LCD_AREGA and LCD_AREGB occur in relation to the clock for the animation frame counter. Because the animation engine cannot know when the writes occur, it is not possible to know whether the A or B register will shift first, which can result in one of the registers shifting twice before the other shifts once.
Affected Conditions / Impacts
Animations that require specific sequencing may not start in the correct state such that frames are not displayed in the correct order.
Workaround
If animation sequences must be seen in a specific order, consider handling this in software instead of using the animation engine. If the purpose of the animation is to denote ongoing activity, use segments that can be cycled in a generic fashion such that the output achieves the desired effect without depending on a specific frame order.
Resolution
There is currently no resolution for this issue.

2.10 LCD_E104 — Increased Current Draw when VLCD > VDDIO and LCD Pins are Used for GPIO

Description of Errata
A leakage path to IOVDD exists when the LCD controller is configured to use the internally boosted or external power supply (LCD_DISPCTRL_VLCDSEL = VEXTBOOST) and VLCD > VDDIO. This is due to PMOS transistors in the LCD pin logic having their source/bulk terminals connected to the highest VDD (thus the LCD power supply when external/boost mode is used) while their gates are connected to IOVDD.
Affected Conditions / Impacts
Use of LCD pins for GPIO results in increased current draw when the LCD controller is configured to use the internally boosted or external supply (LCD_DISPCTRL_VLCDSEL = VEXTBOOST) and VLCD > VDDIO. This is particularly noticeable when the device is operating in EM2 as the LCD to IOVDD supply leakage can amount to tens of microamps. While the GPIO functionality of the LCD pins is not impaired, for certain applications, the increased current draw can be undesirable.
Workaround
Do not use LCD pins for GPIO functionality if the LCD controller is configured to use an external power supply or boost mode, and the resulting VLCD can be greater than the IOVDD supply.
Resolution
There is currently no resolution for this issue.

2.11 PCNT_E102 — PCNT Pulse Width Filtering Does Not Work

Description of Errata
PCNT pulse width filtering does not work.
Affected Conditions / Impacts
The PCNT pulse width filter does not work as intended.
Workaround
Do not use the pulse width filter, i.e., ensure FILT = 0 in PCNTn_CTRL.
Resolution
There is currently no resolution for this issue.

2.12 RTC_E101 — RTC PRS Output

Description of Errata
The RTC PRS output might cause false triggers
Affected Conditions / Impacts
If the RTC is selected as a PRS producer, there might occur glitches which will accidentally cause false triggers.
Workaround
Do not use the RTC as a PRS producer; instead, use one of the other timer sources (e.g. TIMER0).
Resolution
There is currently no resolution for this issue.

2.13 TIMER_E102 — Timer Capture and Debugger

Description of Errata
Timer capture triggered when timer is halted by debugger.
Affected Conditions / Impacts
When DEBUGRUN is disabled and the capture input is HIGH, it is possible to wrongly trigger a capture event by halting the MCU and starting it again (for instance by setting a breakpoint).
Workaround
Enable DEBUGRUN when using a debugger.
Resolution
There is currently no resolution for this issue.

2.14 USART_E101 — U(S)ART Double Buffer

Description of Errata
Transmission control through TXDATAX and TXDOUBLEX does not work with data double buffering.
Affected Conditions / Impacts
When a frame is loaded into the transmission shift register, transmission control bits always taken from outer buffer element. If only one frame is in the U(S)ART buffer, the content of the buffer elements is equivalent, and transmission control bits work as specified. If two frames are in the buffer however, the control bits for the frame in the outer buffer are used for transmitting the frame in inner buffer. This is not a problem for frames consisting of more than 9 bits, since these large frames occupy both the inner and outer buffer elements.
Workaround
If using transmission control bits in registers TXDATAX or TXDOUBLEX, make sure there is not more than one frame in the U(S)ART buffer at a time, or that the control bits are equal. When TXBL in U(S)ARTn_CTRL is cleared, the TXBL status and interrupt flags in U(S)ARTn_STATUS and U(S)ARTn_IF respectively tell when the buffer is empty. When using transmission control bits, a single frame can then be loaded into the USART for transmission.
Resolution
There is currently no resolution for this issue.

2.15 WDOG_E103 — WDOG EM2 Detection with LFXO Digital/Sine Input

Description of Errata
The WDOG will mistake EM2 for EM3 if using LFXO with digital or sine input.
Affected Conditions / Impacts
When the WDOG is using LFXO with digital or sine input as a clock source, it will mistake EM2 for EM3. The EM2RUN and EM3RUN bits of WDOG_CTRL will behave accordingly.
Workaround
When using LFXO with digital/sine input, EM3RUN must be set to keep the WDOG running in EM2.
Resolution
There is currently no resolution for this issue.

3. Resolved Errata Descriptions

This section contains previous errata for EFM32G 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 ACMP_E101 — ACMP Mode

Description of Errata
The ACMP only works in the low power reference mode.
Affected Conditions / Impacts
The ACMP only works in the low power reference mode. When the low power reference mode is disabled (by not setting the LPREF bit in ACMPn_INPUTSEL), the ACMP does not work and its output is always 1.
Workaround
When using the ACMP, put it in its low power reference mode by setting the LPREF bit in ACMPn_INPUTSEL (which is the default setting). In this mode, the power consumption in the reference buffer (VDD and bandgap) is lowered at the cost of accuracy.
Resolution
This issue is resolved in revision C devices.

3.2 ADC_E101 — ADC Temperature Sensor

Description of Errata
The temperature sensor in the ADC does not work.
Affected Conditions / Impacts
The temperature values read when sampling the temperature sensor in the ADC are not correct.
Workaround
Do not use the ADC temperature sensor.
Resolution
This issue is resolved in revision B devices.

3.3 ADC_E102 — ADC SCANGAIN

Description of Errata
SCANGAIN in ADCn_CAL affects the gain setting for single conversions.
Affected Conditions / Impacts
When SCANGAIN and SINGLEGAIN in ADCn_CAL have different values, single conversions will be affected by the SCANGAIN value.
Workaround
Configure SCANGAIN and SINGLEGAIN in ADCn_CAL to the same value. This requires the same reference to be used for both single and scan conversions.
Resolution
This issue is resolved in revision B devices.

3.4 ADC_E104 — ADC 1 Msample/s

Description of Errata
1 Msample/s is not achieved for default ADC bias settings.
Affected Conditions / Impacts
At default ADC bias settings the ADC conversion results are wrong when running the ADC_CLK at 13 MHz, which is required to reach the 1 Msample/s performance. Under typical conditions wrong conversions have been observed for ADC_CLK speeds of 8 MHz and higher.
Workaround
Increase the ADC performance by programming increased ADC bias, for example by using value 0xF0F for register ADCn_BIA-SPROG.
Resolution
This issue is resolved in revision C devices.

3.5 ADC_E105 — ADC Output

Description of Errata
The ADC does not always sample a voltage at (or close to) the middle of its range correctly (e.g. when sampling 1.25V when using the 2.5V internal reference).
Affected Conditions / Impacts
When the ADC is sampling voltages at (or close to) the middle of its range, the ADC output code can be off by a large value (e.g. returning value 1023 or 3072 instead of the expected value of 2048). This effect happens for all ADC reference selections.
Workaround
Perform multiple (e.g. 3) ADC measurements for each ADC sample required and use the median value. Do not average the ADC results, throw away the 1023 or 3072 sample instead.
Resolution
This issue is resolved in revision B devices.

3.6 ADC_E106 — ADC Reference Settling

Description of Errata
The ADC internal references, i.e. 1V25, 2V5 and VDD have a settling time of about 500 µs.
Affected Conditions / Impacts
Measurements done using one of the internal references will not be correct before the reference has settled. This effect appears when switching between references and when the references have been off between samples.
Workaround
When using the internal references, set WARMUPMODE=3 in ADCn_CTRL and wait until the references have settled before taking the first sample.
Resolution
This issue is resolved in revision C devices.

3.7 ADC_E108 — ADC Temperature Sensor

Description of Errata
The temperature sensor in the ADC does not work out of reset.
Affected Conditions / Impacts
Temperature measurements done using the temperature sensor in the ADC will be wrong without the fix described below.
Workaround
To enable the temperature sensor, set *0x400C6018 = 0x6. This fix can not be used at the same time as the fix for CMU_E104.
Resolution
This issue is resolved in revision C devices.

3.8 ADC_E110 — ADC VDD Reference Gives Half Resolution

Description of Errata
When using the internal VDD reference, the ADC resolution is reduced to 11 bits.
Affected Conditions / Impacts
Measurements done with the VDD reference will appear to have been divided by 2.
Workaround
Double each measurement to give the measurements the correcamplitude, sacrificing one bit of resolution. This workaround will not be compatible with devices of later revisions where this erratum is corrected.
Resolution
This issue is resolved in revision C devices.

3.9 ADC_E111 — ADC References Doubled

Description of Errata
In single-ended mode, external and differential references are doubled internally.
Affected Conditions / Impacts
The reference doubling results in a decrease of the ADC resolution by one bit. As an example, when using an external reference of 1 V in single ended mode, a signal with values from 0 V to 1 V will result in adc codes from 0 to 2047 instead of the full 0 to 4095.
Workaround
A temporary fix for the external references is to halve the reference voltage. This will give full resolution, but will not be compatible with devices of later revisions where this erratum has been corrected.
Resolution
This issue is resolved in revision C devices.

3.10 ADC_E112 — ADC Accuracy

Description of Errata
The ADC does not meet the specified accuracy of 11.7 effective bits. The ADC is monotonic and although within specification, the hit frequency for some codes such as 3071 and 3072 is such that DNL is distinctly different from the average DNL (but there are no missing codes). The gain of the 2XVDD reference is larger than 1.0.
Affected Conditions / Impacts
The ADC accuracy may vary depending on the ADC configuration and may in some cases be down to 10 effective bits. The DNL/INL anomalies will cause low level spurs in the output spectra. The gain error for the 2XVDD reference is over 20 LSBs and gain for this reference cannot be calibrated.
Workaround
ADC accuracy can be increased by using hardware oversampling to increase resolution and/or by increasing the ADC bias current. The ADC oversampling rate can be programmed in the OVSRSEL field of the ADCn_CTRL register; the oversampling can be enabled by using the OVS value in the RES field of the ADCn_SINGLECTRL or ADCn_SCANCTRL register. The ADC bias current can be programmed via the ADCn_BIASPROG register.
Resolution
This issue is resolved in revision C devices.

3.11 ADC_E113 — ADC Variability

Description of Errata
The PSRR, CMRR and variability over temperature for the ADC do not meet the specification.
Affected Conditions / Impacts
With a DC input voltage, the ADC output will vary depending on changes in VDD level, input common mode level, and temperature. The temperature related variation particularly applies to the 5VDIFF reference. For a DC level input, variability of the ADC output code over VDD is 2, 32, 7 LSBs for the VDD, 5VDIFF, and external references respectively. With external references, there is about 8 LSBs variation in ADC output code depending on the input common mode level.
Workaround
With external references, adjust the common mode level of the differential external reference to a lower level. For the external differential references, performance is better with the lower external reference negative input level at 0V.
Resolution
This issue is resolved in revision C devices.

3.12 ADC_E114 — Incorrect ADC Calibration Register Reset Value

Description of Errata
The ADC calibration register (ADCn_CAL) are not updated with calibration values from production test during reset.
Affected Conditions / Impacts
The ADC will be uncalibrated out of reset and can show offset and gain error outside of specification.
Workaround
Copy the gain and offset values for the selected ADC reference from the Device Information (DI) page in flash to the corresponding ADCn_CAL register fields before starting a conversion.
Resolution
This issue is resolved in revision C devices.

3.13 ADC_E115 — Incorrect ADC Temperature Sensor Calibration Data

Description of Errata
The ADC temperature sensor calibration value stored in the DI page is not correct.
Affected Conditions / Impacts
Devices with PROD_REV values of 9 or 10 does not have correct ADC temperature sensor reading stored in the ADC0_TEMP_0_READ_1V25 register of the Device Information Page, and using this value for calculating the temperature will yield wrong results.
Workaround
Instead of using the value stored in the Device Information table, use ADC0_TEMP_0_READ_1V25 = 0x906 and CAL_TEMP_0 = 0x19 when calculating the temperature. These values are gathered from production data, and will give an accuracy where 3x the standard deviation correspond to 5.2 degrees celsius.
Resolution
This issue is resolved in revision D devices.

3.14 BOD_E101 — BOD Threshold

Description of Errata
The Brown-Out Detector (BOD) threshold voltage is calibrated to a too high value.
Affected Conditions / Impacts
The high BOD threshold voltage may create sporadic BOD resetswhile the EFM32 is running in Energy Mode 2. Also, the BOD may cause a reset at higher voltages than specified as the threshold voltage in the Electrical Characterisitics.
Workaround
Download Development Kit Board Support Library and Example Code (rev 1.1.1 or later) and include `efm32_chip.h` in your project. In the start of the application code, call `void CHIP_Init(void)`. This procedure will re-program the device to a safe BOD threshold.
Resolution
This issue is resolved in revision B devices.

3.15 CMU_E101 — Peripheral Clocks Active In EM2/EM3 During Debug

Description of Errata
When a debug session has been active since the last reset, EM1 is entered when trying to enter EM2 or EM3.
Affected Conditions / Impacts
The device cannot enter EM2/3 if a debug session has been entered since the last reset. When attempting to go to either EM2 or EM3, the system goes to EM1, and the peripheral clocks keep running, even though they should have been turned off in EM2/EM3. This is only an issue when debugging a system.
Workaround
If the debugger is running, clear HFPERCLKEN in CMU_HFPERCLKDIV before going to EM2/EM3 and set it when going back to EM0.
Resolution
This issue is resolved in revision B devices.

3.16 CMU_E102 — LFRCO/HFRCO Frequency Change During EM2/3

Description of Errata

RCO oscillator frequency can become unstable on transitions between EM2/3 and EM0.

Affected Conditions / Impacts

When switching between EM0 and EM2/3, the following events can happen occasionally:

1. The frequency of LFRCO becomes off by up to 14%.
2. The frequency of HFRCO becomes off by up to 6%.

The frequency will be off for a shorter or longer period.

Workaround

Make this line of code part of your startup code, typically in the start of `main()`:

```
*(volatile unsigned int*) 0x400C600CUL = 0x00020100;
```

As a result of this workaround, the current consumption in EM2/3 will go up by 450 nA. This fix is not compatible with devices of later revisions where this erratum has been corrected.

Resolution

This issue is resolved in revision B devices.

3.17 CMU_E103 — Wrong RCO Frequency

Description of Errata

The HFRCO, AUCHFRCO and LFRCO oscillators has wrong frequency when running with default settings.

Affected Conditions / Impacts

The oscillator frequency has not been programmed with correct calibration values, and the frequencies are not within the expected frequency ranges.

Workaround

The oscillator frequency can be calibrated in the Clock Management Unit, which is described in the CMU chapter of the EFM32G Reference Manual.

Resolution

This issue is resolved in revision B devices.

3.18 CMU_E104 — Energy Mode Transitions Cause HFRCO Overshoot

Description of Errata
Transitions between energy modes may cause an overshoot in the HFRCO frequency.
Affected Conditions / Impacts
When switching between energy modes, there is a slight chance that HFRCO will temporarily overshoot its configured frequency and in some cases cause a BOD reset. This overshoot may be up to 50% and may take the system out of its allowed operating conditions by having a system clock higher than 32 MHz. Note that when the MSC is configured to use zero waitstates when accessing flash, the maximum core frequency is 16 MHz.
Workaround
To workaround this issue, set *0x400C6018 = 0xC201. This gives better HFRCO stability at the cost of an increased current consumption in EM0 and EM1 of 10 uA. This fix is not compatible with the fix for ADC_E108. This fix is not compatible with devices of later revisions where this erratum has been corrected.
Resolution
This issue is resolved in revision C devices.

3.19 CMU_E105 — AUXHFRCO Active in EM2/EM3

Description of Errata
AUXHFRCO is not disabled automatically when entering EM2/EM3.
Affected Conditions / Impacts
If AUXHFRCO is running while in EM2/EM3 and the EMVREG bit in EMU_CTRL is cleared, this may result in an unstable system.
Workaround
Disable AUXHFRCO by writing a 1 to AUXHFRCODIS in CMU_OSCENCMD before going to EM2/EM3. When waking up, enable the AUXHFRCO again if needed by writing a 1 to AUXHFRCOEN in CMU_OSCENCMD.
Resolution
This issue is resolved in revision B devices.

3.20 CMU_E107 — Disabled Low Frequency Clocks

Description of Errata
Disabled Low Frequency Clocks tick once every second.
Affected Conditions / Impacts
The LFA and LFB clocks tick with a frequency equal to the source clock divided by 32768 when the correspondingenable bit in CMU_LFACLKEN0/CMU_LFBCLKEN0 is not set. So, for example, if the RTC is enabled and RTC in CMU_LFACLKEN0 is 0,the RTC will tick once every second. Notice that it is not possible to write to a Low Energy Peripheral when the clock tothe peripheral is not enabled. Thus, the effects of this issue is only seen when the clock to an active Low Energy Peripheralis turned off.
Workaround
Disable the Low Energy Peripheral before disabling the clock to the Low Energy Peripheral to avoid unexpected behaviour.
Resolution
This issue is resolved in revision C devices.

3.21 CMU_E108 — LFxCLKEN Write

Description of Errata
First write to LFxCLKEN can be missed.
Affected Conditions / Impacts
For devices with PROD_REV < 15, enabling the clock for LFA/LFB after reset and then immediately writing LFACLKEN/LFBCLKEN may cause the write to miss its effect.
Workaround
For devices with PROD_REV < 15, make sure CMU_SYNCBUSY is not set before writing LFACLKEN/LFBCLKEN. Can temporarily switch to HFCORECLKLEDIV2 to speed up clearing synchbusy.
Resolution
This issue is resolved in revision D devices.

3.22 CMU_E109 — LFXO Configuration Incorrect

Description of Errata
LFXO configuration incorrect.
Affected Conditions / Impacts
For devices with PROD_REV < 15, the default value for LFXOBOOST in CMU_CTRL is wrong.
Workaround
On devices with PROD_REV < 15, change LFXOBOOST to 0.
Resolution
This issue is resolved in revision D devices.

3.23 DAC_E101 — DAC Sample-Hold

Description of Errata
When the DAC is in sample/hold mode, the DAC output is not correctly held, but drifts faster than specified.
Affected Conditions / Impacts
The DAC output starts drifting in the order of 10 mV/μs after two DAC clock cycles.
Workaround
Put the DAC in continuous mode by setting the CONVMODE field in the DACn_CTRL register to CONTINUOUS. The DAC channels will then drive their outputs continuously with the data in the DACn_CHxDATA registers. This mode will maintain the output voltage and refresh is therefore not needed. As the DAC cores are not turned off between samples in continuous mode, the power consumption is somewhat increased compared to sample/hold mode.
Resolution
This issue is resolved in revision C devices.

3.24 DAC_E102 — DAC Enabling

Description of Errata
DAC conversions done closely after enabling the DAC channel are incorrect.
Affected Conditions / Impacts
The DAC output takes about 600 µs (under typical conditions) to settle after a DAC channel has been enabled via setting field CH0EN in DACn_CH0CTRL (or CH1EN in DACn_CH1CTRL for channel 1). The effect is most visible for the 1.25V and 2.5V internal references.
Workaround
After enabling a DAC channel, wait 600 µs before programming the channel data (via DACn_CH0DATA, DACn_CH1DATA, or DACn_COMBDATA).
Resolution
This issue is resolved in revision C devices.

3.25 DAC_E103 — DAC Ringing

Description of Errata
Ringing effects can be observed on the DAC output.
Affected Conditions / Impacts
When applying large steps to the DAC, ringing can be observed on the output (up to 100 mV peak-to-peak depending on load). The oscillations will last no longer than 1 µs.
Workaround
Add a filter on the DAC output or don't apply large steps.
Resolution
This issue is resolved in revision C devices.

3.26 DAC_E104 — DAC Sample-Hold/Sample-Off

Description of Errata
When a sample/refresh is done while in the sample-hold and sample-offmodes, a dip in the DAC output voltage occurs.
Affected Conditions / Impacts
The voltage-dip causes noise.
Workaround
Use the DAC in continuous mode by setting the CONVMODE field in the DACn_CTRL register to CONTINUOUS.
Resolution
This issue is resolved in revision C devices.

3.27 DAC_E105 — DAC Startup

Description of Errata
When enabling a DAC channel, there may be a transient on the channel output. This transient may be up to 800 mV and last about 1 µs on an unloaded DAC.
Affected Conditions / Impacts
The DAC output is incorrect a short while after enabling the DAC channel.
Workaround
To prevent the transient on the DAC output, make sure the DAC output is disabled by clearing OUTMODE in DACn_CTRL when enabling a DAC channel.
Resolution
This issue is resolved in revision C devices.

3.28 DAC_E107 — DAC Accuracy

Description of Errata
The DAC does not meet the specified accuracy of 11.5 effective bits. The DAC linearity does not meet the specification for all input codes.
Affected Conditions / Impacts
The DAC accuracy may vary depending on the DAC configuration and may in some cases be down to 9 effective bits (which is primarily caused by the SFDR without external filtering being limited to about 55 dB in case of the 1V25 reference). For some input codes, e.g. 1024, 2048, 3072, an increase/decrease of one input code will result in an output change equal to up to 3 input codes.
Workaround
No workaround.
Resolution
This issue is resolved in revision C devices.

3.29 DAC_E108 — Incorrect DAC Calibration Register Reset Value

Description of Errata
The DAC calibration register (DACn_CAL) is not updated with calibration values from production test during reset.
Affected Conditions / Impacts
The DAC will be uncalibrated out of reset and can show offset and gain error outside of specification.
Workaround
Copy the gain and offset values for the selected DAC reference from the Device Information (DI) page in flash to the corresponding DACn_CAL register fields before starting a conversion.
Resolution
This issue is resolved in revision C devices.

3.30 EBI_E101 — EBI Lines

Description of Errata
When the EBI is idle, the 16 address/data lines are left floating.
Affected Conditions / Impacts
The floating address/data lines may cause an increase in current consumption since the schmitt triggers of the GPIO inputs are still on.
Workaround
Apply external pull-ups to the address-data lines, or make sure to control the address/data lines from software when not used.
Resolution
This issue is resolved in revision C devices.

3.31 EMU_E101 — EM Transition Brown Out

Description of Errata
In rare situations, transitioning between energy modes may cause a Brown Out (BO).
Affected Conditions / Impacts
When switching between energy modes, there is a chance that the system will experience a BO. In that case, the system will be reset and the error condition can be detected by reading the RMU_RSTCAUSE register, which will then show that an internal BO was the reason for the reset.
Workaround
To fix issue, set *0x400C6020 |= 0x6000. This prevents the BO, but results in an increase of current consumption in EM0 and EM1 by about 4%. This fix is not compatible with devices of later revisions where this erratum has been corrected.
Resolution
This issue is resolved in revision C devices.

3.32 EMU_E102 — DMA Clock EM2/EM3

Description of Errata
When the DMA clock is disabled, the EFM32 is not able to go to Energy Modes 2 or 3.
Affected Conditions / Impacts
The DMA will prevent the system to go to EM2/EM3 as long as the DMA clock is disabled.
Workaround
Make sure the DMA clock is enabled when going to EM2/EM3. The DMA clock can be enabled in the CMU.
Resolution
This issue is resolved in revision B devices.

3.33 EMU_E105 — Debug Unavailable During DMA Processing from EM2

Description of Errata
The debugger cannot access the system processing DMA request from EM2.
Affected Conditions / Impacts
DMA requests from the LEUART can trigger a DMA operation from EM2. While waiting for the DMA to fetch data from the respective peripheral, the debugger cannot access the system. If such a DMA request is not handled by the DMA controller, the system will keep waiting for it while denying debug access.
Workaround
Make sure DMA requests triggered from EM2 are handled.
Resolution
This issue is resolved in revision D devices.

3.34 EMU_E106 — SWO Line Pulled Low in EM2

Description of Errata
SWO pulled low in EM2.
Affected Conditions / Impacts
The SWO line is pulled low in EM2. This can be interpreted as garbage by an outside observer.
Workaround
Before entering EM2, disable pin-enable by clearing SWOPEN in GPIO_ROUTE and set the SWO pin output high. After exiting EM2, the SWO pin should be re-enabled.
Resolution
This issue is resolved in revision D devices.

3.35 EMU_E107 — Interrupts During EM2 Entry

Description of Errata
An interrupt from a peripheral running from the high frequency clock that is received during EM2 entry will cause the EMU to ignore the SLEEPDEEP flag.
Affected Conditions / Impacts
During EM2 entry, the high frequency clocks that are disabled during EM2 will run for some clock cycles after WFI is issued to allow safe shutdown of the peripherals. If an enabled interrupt is requested from one of these non-EM2 peripherals during this shutdown period, the attempt to enter EM2 will fail, and the device will enter EM1 instead. As a result, the pending interrupt will immediately wake the device to EM0.
Workaround
Before entering EM2, disable all high frequency peripheral interrupts in the core.
Resolution
This issue is resolved in revision E devices.

3.36 HFRCO_E101 — HFRCO Calibration

Description of Errata
The Device Information page does not contain calibration values for the 1 MHz, 7 MHz, 11 MHz, and 21 MHz frequency bands.
Affected Conditions / Impacts
The HFRCO frequency will be outside the expected frequency range when applying the calibration value from the device information page.
Workaround
The oscillator frequency can be calibrated in the Clock Management Unit, which is described in the CMU chapter of the EFM32G Reference Manual.
Resolution
This issue is resolved in revision B devices.

3.37 I2C_E101 — I^{2}C RX Overflow

Description of Errata
If reception of a byte by the RX shift register is completed while there is still a byte in the RX buffer, the byte in the shift register is silently discarded.
Affected Conditions / Impacts
If a received byte is acknowledged before it is read out of the RXDATA, all new bytes received before the read operation are discarded. A new byte is not discarded if the read operation is performed before the new byte is fully received.
Workaround
Make sure to read the RX buffer before the reception of the next byte completes. One way to ensure this is to always read a received byte before acknowledging it.
Resolution
This issue is resolved in revision B devices.

3.38 I2C_E102 — I^{2}C Disabled After EM2/EM3

Description of Errata
If the USART0 clock is disabled, the I2C will not work when waking up from EM2/EM3.
Affected Conditions / Impacts
When waking up from EM2/EM3 with the USART0 clock disabled, the I2C module will be in a disabled state until the USART0 clock has been enabled again.
Workaround
Make sure the USART0 clock is enabled when using the I2C. Alternatively, enable the USART0 clock for a short while after exiting EM2/EM3.
Resolution
This issue is resolved in revision C devices.

3.39 LCD_E101 — LCD Com Line

Description of Errata
Artifacts are seen on the LCD com lines (LCD_COM0-LCD_COM3).
Affected Conditions / Impacts
The LCD com lines (LCD_COM0-LCD_COM3) show intermediate voltage levels before settling to their correct values. The artifact depends on the LCD contrast settings.
Workaround
Always adjust the LCD contrast relative to VDD (VLCD) by setting CONCONF field to 0 in LCD_DISPCTRL. This is its default value. Use any contrast level other than the maximum value, so keep CONLEV in LCD_DISPCTRL in the range 0-30. The default value of CONLEV is okay.
Resolution
This issue is resolved in revision B devices.

3.40 LCD_E102 — LCD Voltage Boost Current

Description of Errata
When the LCD boost target voltage is close to or lower than VDD, the voltage boost function draws an excessive amount of current
Affected Conditions / Impacts
If voltage boost is enabled when the boost target voltage, given by VBLEV in LCD_DISPCTRL is lower than or close to VDD, the current consumption of the LCD driver increases by about 500 μA.
Workaround
Make sure the voltage boost is not enabled before VDD is below the boost target voltage. This can be done by using the VCMP module to monitor VDD and enable/disable voltage boost when VDD goes below/above a given voltage, or by using the ADC to regularly sample VDD.
Resolution
This issue is resolved in revision C devices.

3.41 LEUART_E101 — LEUART + DMA

Description of Errata
EM2 cannot be entered when transmitting the last byte using LEUART and DMA.
Affected Conditions / Impacts
When using the LEUART with DMA in EM2, TXDMAWU in LEUARTn_CTRL must be cleared when the DMA has no more data to transmit. Otherwise the LEUART will keep the system awake waiting for data from the DMA. The way to do this is to clear TXDMAWU in the DMA DONE interrupt for the channel feeding the LEUART with data. In this device revision, the DMA DONE interrupt will not trigger a wakeup from EM2, and software will thus not be able to clear TXDMAWU immediately when a transmission has been completed, causing the system to be awake more than necessary.
Workaround
Use the TX complete interrupt (TXC) in the LEUART to clear TXDMAWU, or clear TXDMAWU in the DMA DONE interrupt and make sure the TXC interrupt is triggered. The system will then be awake with a higher power consumption while the last byte is transmitted by the LEUART, but will be allowed to go back to EM2 once TXDMAWU has been cleared.
Resolution
This issue is resolved in revision B devices.

3.42 LEUART_E102 — LEUART Baudrate

Description of Errata
The LEUART baudrate generator should have an integral baudrate error no larger than ±0.5 clock cycles. However, the error may be up to 1 clock cycle.
Affected Conditions / Impacts
For some baudrate settings, the jitter will be higher than ±0.5 clock cycles, and the average baudrate value will also not be as expected. For ~9600 b/s @ 32.768 kHz oscillator frequency, CLKDIV=0x268 and CLKDIV=0x270 for instance gives significantly different baudrates. For lower baudrates, there should be no problem when using a 32 kHz oscillator.
Workaround
For 9600 b/s @ 32.768 kHz, use CLKDIV=0x270, which gives a baudrate of ~9534 b/s. This workaround will not be compatible with devices of later revisions where this erratum has been corrected.
Resolution
This issue is resolved in revision C devices.

3.43 LEUART_E103 — LEUART RXOF

Description of Errata
In rare situations, the RX overflow interrupt can be set even though RX data is not lost.
Affected Conditions / Impacts
RXOF is set one cycle too early with the consequence that if RX buffer is read in the same internal clock cycle as RXOF is set, the frame causing RX data is actually loaded into the RX buffer. Thus, the overflow interrupt does not guarantee that data has been lost.
Workaround
Consider RXOF as an indication that an overflow might have occurred.
Resolution
This issue is resolved in revision C devices.

3.44 LETIMER_E101 — Buffered Top Value

Description of Errata
CNT is updated with LETIMERn_COMP0 instead of LETIMERn_COMP1 when REP0 goes to zero.
Affected Conditions / Impacts
In Buffered Mode, both CNT and LETIMERn_COMP0 shall be updated with new LETIMERn_COMP1 value when REP0 goes to zero. Instead, CNT gets the previous LETIMERn_COMP0 value, while LETIMERn_COMP0 is correctly updated. Thus, the first period in the next repeat-sequence is the previous top value (LETIMERn_COMP0) and not the new top value (LETIMERn_COMP1) as one would expect.
Workaround
No workaround.
Resolution
This issue is resolved in revision C devices.

3.45 LFRCO_E101 — LFRCO Frequency

Description of Errata
The frequency of the LFRCO changes with up to 30% between EM0/EM1 and EM2/EM3.
Affected Conditions / Impacts
Calibrating the LFRCO for a given frequency in EM0/EM1 will not guarantee the same frequency in EM2/EM3.
Workaround
Use LFXO if an accurate clock frequency is important.
Resolution
This issue is resolved in revision C devices.

3.46 LFXO_E102 — LFXO Temperature Sensitivity

Description of Errata
LFXO may not start.
Affected Conditions / Impacts
On some devices the LFXO may not start, on others the LFXO may stop when temperature approaches -40C. In the latter case, the LFXO will start up again when the temperature rises.
Workaround
Place a resistor in parallel with the LFXO crystal. The resistor should be approximately 50 MΩ.
Resolution
This issue is resolved in revision C devices.

3.47 PCNT_E101 — PCNT0 TOP Register

Description of Errata
The reset value of the TOP register of PCNT0 is incorrect.
Affected Conditions / Impacts
When counting downwards, the pulse counter underflows to an incorrect value. When counting upwards, no interrupt flag is set when counting beyond the maximum value of 0xFF.
Workaround
Before enabling PCNT0, write the desired top value to the TOPB register. Then load this value into the TOP register by setting LTOPBM in the CMD register.
Resolution
This issue is resolved in revision C devices.

3.48 TIMER_E101 — TIMER Up/Down Mode

Description of Errata
In up/down mode, TIMERn_CCV and TIMER_TOP are updated on both overflow and underflow.
Affected Conditions / Impacts
Up/down mode may generate pulses that are not centered around the TIMERn_TOP value.
Workaround
Correct PWM operation can be ensured by updating TIMERn_CCx_CCVB after the timer overflow flag (TIMERn_IF_OF) is set. When using the DMA overflow/underflow trigger to update TIMERn_CCx_CCVB, the CCV samples in the source buffer must duplicated once. For example, if the CCV sequence is {0x24, 0x100, 0x99}, the buffer the DMA reads from must be {0x24, 0x24, 0x100, 0x100, 0x99, 0x99}. The first sample is written on the underflow event, and the second (duplicate) is written on the overflow event. The DMA work-around is not compatible with the revision C fix.
Resolution
This issue is resolved in revision C devices.

3.49 USART_E102 — U(S)ART RXOF

Description of Errata
In rare situations, the RX overflow interrupt can be set even though RX data is not lost.
Affected Conditions / Impacts
RXOF is set one cycle too early with the consequence that if RX buffer is read in the same internal clock cycle as RXOF is set, the frame causing RX data is actually loaded into the RX buffer. Thus, the overflow interrupt does not guarantee that data has been lost. In addition, this frame will have its LSB cleared when transmitting LSB first, and its MSB cleared if MSB is transmitted first.
Workaround
On RX overflow, the last frame received may contain a bit error. Disregard this frame.
Resolution
This issue is resolved in revision C devices.

3.50 USART_E103 — U(S)ART Slave TXUF Causes Shift

Description of Errata
Slave TXUF may take TX out of sync with master.
Affected Conditions / Impacts
When in sync slave mode, an underflow may take the slave TX out of sync with the master clock, and data received at the master will be shifted.
Workaround
Make sure the TX does not underflow when in slave mode.
Resolution
This issue is resolved in revision C devices.

3.51 USART_E104 — U(S)ART AUTOTRI One Cycle Late

Description of Errata
The AUTOTRI feature enables output one cycle late.
Affected Conditions / Impacts
When output enable is controlled by AUTOTRI, the output will be enabled one clock cycle after the first data is output. The largest effect of this will occur when the U(S)ART clock is close to the U(S)ART baudrate.
Workaround
If the timing of AUTOTRI is not sufficient, use the TXTRIEN/TXTRIDIS commands to enable and disable the USART output.
Resolution
This issue is resolved in revision C devices.

3.52 USART_E105 — U(S)ART Fractional Baudrate

Description of Errata
Fractional baudrate is wrong for divisors 1.25, 1.50, and 1.75.
Affected Conditions / Impacts
The fractional baudrate generated by the U(S)ART is wrong for the divisors 1.25, 1.50, and 1.75. This corresponds to the CLKDIV values 0x40, 0x80, and 0xC0.
Workaround
Avoid using the erroneous divisors.
Resolution
This issue is resolved in revision C devices.

3.53 USART_E106 — U(S)ART Slave TX Tristate

Description of Errata
TXTRIEN/TXTRIDIS do not work in half-duplex synchronous slave mode.
Affected Conditions / Impacts
In slave mode, the TXTRIEN and TXTRIDIS commands have no effect on the enabled state of the MISO output. The MISO output is controlled solely by whether the slave is selected by the master or not. This affects half-duplex communication when using the USART in synchronous slave mode.
Workaround
To explicitly control output-enable in slave mode, use GPIO to control the mode of the MISO pin.
Resolution
This issue is resolved in revision C devices.

3.54 USART_E107 — U(S)ART TXC

Description of Errata
TXC may be set even though data is in the TX buffer.
Affected Conditions / Impacts
If data is written to the U(S)ART at the same time as the previous transmission completes, the TXC status flag may be set even though new data has been written to the USART and the USART starts transmission of this data.
Workaround
Qualify the TXC status with the status of TXBL to know whether the U(S)ART is idle and empty.
Resolution
This issue is resolved in revision C devices.

3.55 USART_E108 — U(S)ART Slave TX Data Required Early

Description of Errata
For CLKPHA=0, slave TX data is required too early.
Affected Conditions / Impacts
When operating with CLKPHA=0 and the slave TX is empty when CS is asserted or on the last edge of a frame, the slave underflows. In this case, no data will be clocked out on the next frame.
Workaround
Make sure the slave TX has data in time or use CLKPHA=1 in synchronous slave mode.
Resolution
This issue is resolved in revision C devices.

3.56 USART_E109 — U(S)ART Slave TXUF Artifacts

Description of Errata
MISO toggling and TXUF set multiple times on slave TX underflow.
Affected Conditions / Impacts
When underflowing, a U(S)ART slave may give a pulse on MISO on every setup-edge of the SPI clock. The underflow interrupt flag may also be set multiple times during a frame where the slave underflows.
Workaround
Make sure the slave TX has data in time.
Resolution
This issue is resolved in revision C devices.

3.57 USART_E110 — U(S)ART Slave TX Data Lost

Description of Errata
For CLKPHA=0, a frame is cleared from TX buffer on CS deassert.
Affected Conditions / Impacts
When operating as a slave with CLKPHA=0, a frame is cleared from the TX buffer when CS is asserted by the SPI master.
Workaround
Use CLKPHA=1 or make sure to not write more data to the slave TX than what is intended to transmit during a transmission.
Resolution
This issue is resolved in revision C devices.

3.58 USART_E111 — U(S)ART RX DMA Request After TX

Description of Errata
For CLKDIV less than 0x100 RX DMA double requests come after the TX DMA double request.
Affected Conditions / Impacts
When operating with LOOPBK=1 writing and reading 16-bits at a time to the USART using DMA on a loaded system, this will result in TX requests being handled before RX requests, which may result in TX transmitting frames too fast for RX to handle, leading to RX overflows.
Workaround
Keep the system load under control and handle the U(S)ART overflows.
Resolution
This issue is resolved in revision C devices.

3.59 VCMP_E101 — VCMP Mode

Description of Errata
The VCMP only works in the low power reference mode.
Affected Conditions / Impacts
The VCMP only works in the low power reference mode. When the low power reference mode is disabled (by not setting the LPREF bit in VCMP_INPUTSEL), the VCMP does not work and its output is always 1.
Workaround
When using the VCMP, put it in its low power reference mode by setting the LPREF bit in VCMP_INPUTSEL (which is the default setting). In this mode, the power consumption in the reference buffer (VDD and bandgap) is lowered at the cost of accuracy.
Resolution
This issue is resolved in revision C devices.

3.60 VCMP_E102 — VCMP Current

Description of Errata
The current consumption of the VCMP is too high in low power reference mode.
Affected Conditions / Impacts
When the VCMP is enabled in low power reference mode, the current consumption is higher than specified. The current is the same independent of whether the low power reference mode is enabled or not.
Workaround
No workaround.
Resolution
This issue is resolved in revision C devices.

3.61 WDOG_E101 — WDOG in EM2/EM3

Description of Errata
When the watchdog (WDOG) triggers a reset while the EFM32 is in EM2 or EM3, the resulting behaviour is undefined.
Affected Conditions / Impacts
After a watchdog reset from EM2/EM3, the EFM32 may go directly to hard fault or may not start at all.
Workaround
Do not use the watchdog in EM2/EM3 by disabling WDOG before entering EM2/EM3. Note that the EM2RUN and EM3RUN bits cannot be used (see WDOG_E102 erratum description).
Resolution
This issue is resolved in revision B devices.

3.62 WDOG_E102 — WDOG Does Not Freeze in EM2/EM3

Description of Errata
The WDOG keeps running in EM2 and EM3, even though the EM2RUN and EM3RUN bits in WDOG_CTRL are programmed to 0.
Affected Conditions / Impacts
If the WDOG is enabled when entering EM2 or EM3, a WDOG reset will occur unless the system wakes up from EM2/EM3 (by interrupt) and clears the WDOG timer before the WDOG times out.
Workaround
Disable WDOG before entering EM2/EM3 by writing EN bit in WDOG_CTRL to 0. This requires that the WDOG configuration is unlocked (LOCK bit in WDOG_CTRL = 0). If the WDOG configuration is locked, the WDOG will remain enabled in EM2/EM3, and the system must wake up the device from EM2/EM3 and clear the WDOG before the WDOG times out.
Resolution
This issue is resolved in revision D devices.

4. Revision History

Revision 2.30

March, 2021

- Added CMU_E115.
- Added LCD_E103, LCD_E104.
- Migrated to new errata document format.

Revision 2.20

March 1st, 2017

- Added ADC_E118, DAC_E109, EBI_E101, and PCNT_E102 .
- Moved EMU_E107 to fixed for revision E devices.
- Updated errata formatting.
- Merged all errata documents for EFM32G devices into one document.
- Merged errata history and errata into one document.

Revision 2.10

February 20th, 2015

- Added PCNT_E102 and DAC_E109.
- Added EMU_E107 for all devices except EFM32G800.
- Corrected note on external driver in EMU_E103 for all devices except EFM32G800.
- Updated link to errata for older revisions for all devices except EFM32G800.

Revision 2.00

August 21st, 2013

- Updated disclaimer, trademark and contact information.

Revision 1.90

July 30th, 2013

- Added DMA_E101.
- Updated errata naming convention.

Revision 1.80

December 11th, 2012

- Removed erratas no longer present for chip revision D: ADC_E115, CMU_E108, CMU_E109, EMU_E105, EMU_E106, WDOG_E102.
- Added AES_E101 and TIMER_E102.

Revision 1.70

January 11th, 2011

- Added CMU_E109.
- Updated CMU_E108.

Revision 1.60

November 10th, 2011

- Added CMU_E108 and EMU_E106.

Revision 1.50

May 20th, 2011

- Added ADC_E115, EMU_E105, and WDOG_E103.

Revision 1.40

November 17th, 2010

- Added EMU_E104.

Revision 1.30

October 26th, 2010

- Added EMU_E103 and RTC_E101.

Revision 1.20

August 31st, 2010

- Removed errata not valid for chip revision C.
- Added WDOG_E102.

Revision 1.10

June 25th, 2010

- Removed ADC_E107, DAC_E106, and LCD_E103.
- Added ACMP_E101, ADC_E112, ADC_E113, CMU_E106, CMU_E107, DAC_E107, LEUART_E103, LETIMER_E101, TIMER_E101, USART_E102, USART_E103, USART_E104, USART_E105, USART_E106, USART_E107, USART_E108, USART_E109, USART_E110, USART_E111, VCMP_E101, and VCMP_E102.

Revision 1.00

April 23rd, 2010

- Removed ADC_VCM errata.
- Updated the errata which are to be fixed in chip revision C.

May 23th, 2011

- Initial preliminary release for EFM32G222 devices.

October 15th, 2014

- Initial release for EFM32G800 devices.

Revision 0.90

July 15th, 2011

- Initial preliminary release for EFM32G232 and EFM32G842 devices.

Revision 0.80

April 8th, 2010

- Initial preliminary release for EFM32G200, EFM32G210, EFM32G230, EFM32G280, EFM32G290, EFM32G840, EFM32G880, and EFM32G890 devices.

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

www.silabs.com