



# Mighty Gecko Mesh Networking Module

## MGM12P Errata

---



This document contains information on the errata of MGM12P. The latest available revision of this device is revision V2.

For errata on older revisions, please refer to the errata history section for the device. The revision information is typically specified in or near the trace code on the device. Refer to the package marking information in the data sheet for more information.

Errata effective date: June 12th, 2017.

## 1. Active Errata Summary

These tables lists all known errata for the MGM12P and all unresolved errata in revision V2 of the MGM12P.

**Table 1.1. Errata History Overview**

Designator	Title/Problem	Exists on Re- vision:
		V2
ADC_E213	ADC KEEPINSLOWACC Mode	X
ADC_E224	ADC Warm-Up Ready Can Cause IDAC, ACMP, or CSEN to Not Function	X
CUR_E203	Occasional Extra EM0/1 Current	X
DBG_E204	Debug Recovery with JTAG Does Not Work	X
DCDC_E204	Potential Brownout when Enabling the DCDC from Off to Low Power Mode	X
EMU_E212	Delay Required Between Successive Voltage Scaling Commands	X
EMU_E214	Device Erase Cannot Occur if Voltage Scaling Level is Too Low	X
RMU_E202	External Debug Access Not Available After Watchdog or Lockup Full Reset	X
RMU_E203	AVDD Ramp Issue	X
RTCC_E203	Potential Stability Issue with RTCC Registers	X
RTCC_E204	Disabling the RTCC Backup RAM may Consume Extra Current	X
RTCC_E205	Wrap Event Can Be Missed	X
USART_E201	USART DMA Transactions Fail with Slow Peripheral Clocks	X
VDAC_E201	VDAC Output Drives All APORT Buses Simultaneously	X
VDAC_E202	PRS Outputs Not Generated when Interrupt Flag is Set	X

**Table 1.2. Active Errata Status Summary**

Errata #	Designator	Title/Problem	Workaround	Affected Revision	Resolution
			Exists		
1	ADC_E213	<a href="#">ADC KEEPINSLOWACC Mode</a>	No	V2	—
2	ADC_E224	<a href="#">ADC Warm-Up Ready Can Cause IDAC, ACMP, or CSEN to Not Function</a>	Yes	V2	—
3	CUR_E203	<a href="#">Occasional Extra EM0/1 Current</a>	No	V2	—
4	DBG_E204	<a href="#">Debug Recovery with JTAG Does Not Work</a>	Yes	V2	—
5	DCDC_E204	<a href="#">Potential Brownout when Enabling the DCDC from Off to Low Power Mode</a>	Yes	V2	—
6	EMU_E212	<a href="#">Delay Required Between Successive Voltage Scaling Commands</a>	Yes	V2	—
7	EMU_E214	<a href="#">Device Erase Cannot Occur if Voltage Scaling Level is Too Low</a>	Yes	V2	—
8	RMU_E202	<a href="#">External Debug Access Not Available After Watchdog or Lockup Full Reset</a>	Yes	V2	—

Errata #	Designator	Title/Problem	Workaround Exists	Affected Revision	Resolution
9	RMU_E203	<a href="#">AVDD Ramp Issue</a>	Yes	V2	future V2 module update
10	RTCC_E203	<a href="#">Potential Stability Issue with RTCC Registers</a>	Yes	V2	—
11	RTCC_E204	<a href="#">Disabling the RTCC Backup RAM may Consume Extra Current</a>	Yes	V2	—
12	RTCC_E205	<a href="#">Wrap Event Can Be Missed</a>	Yes	V2	—
13	USART_E201	<a href="#">USART DMA Transactions Fail with Slow Peripheral Clocks</a>	No	V2	—
14	VDAC_E201	<a href="#">VDAC Output Drives All APORT Buses Simultaneously</a>	Yes	V2	—
15	VDAC_E202	<a href="#">PRS Outputs Not Generated when Interrupt Flag is Set</a>	Yes	V2	—

## 2. Detailed Errata Descriptions

### 2.1 ADC\_E213 – ADC KEEPINSLOWACC Mode

<b>Description of Errata</b>
When WARMUP-MODE in ADCn_CTRL is set to KEEPINSLOWACC, the ADC does not track the input voltage. Also, the ADC keeps the input muxes closed even during channel switching, making it not recommended to operate the ADC in KEEPINSLOWACC mode.
<b>Affected Conditions / Impacts</b>
KEEPINSLOWACC warmup mode does not function properly.
<b>Workaround</b>
There is currently no workaround for this issue.
<b>Resolution</b>
There is currently no resolution for this issue.

### 2.2 ADC\_E224 – ADC Warm-Up Ready Can Cause IDAC, ACMP, or CSEN to Not Function

<b>Description of Errata</b>
The IDAC, ACMP, or CSEN modules use the warm up timing module in the ADC to determine when the peripherals are ready for use. However, if the ADC is enabled first, this timing module can fail to properly handshake with a low probability, causing the IDAC, ACMP, or CSEN modules to never finish warming up. The ADC is not affected by this issue and will always be available after it is enabled.
<b>Affected Conditions / Impacts</b>
Systems using the IDAC, ACMP, or CSEN modules in conjunction with the ADC can see intermittent failures where these modules do not operate.
<b>Workaround</b>
To work around this issue, enable the IDAC, ACMP, or CSEN modules before enabling the ADC. This will ensure the handshaking logic between the ADC and other modules functions correctly.
<b>Resolution</b>
There is currently no resolution for this issue.

### 2.3 CUR\_E203 – Occasional Extra EM0/1 Current

<b>Description of Errata</b>
Occasionally when exiting EM2, a low voltage oscillator sometimes continues to run and causes the device to draw an extra ~10 $\mu$ A when in EM0 or EM1. This oscillator automatically resets when entering EM2 or EM3, so the extra current draw is not present in these modes.
<b>Affected Conditions / Impacts</b>
Systems using EM2 may occasionally see an extra ~10 $\mu$ A of current draw in EM0 or EM1.
<b>Workaround</b>
There is currently no workaround for this issue.
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.4 DBG\_E204 – Debug Recovery with JTAG Does Not Work

<b>Description of Errata</b>
The debug recovery algorithm of holding down pin reset, issuing a System Bus Stall AAP instruction, and releasing the reset pin does not work when using the JTAG debug interface. When using the JTAG debug interface, the core will continue to execute code as soon as the reset pin is released.
<b>Affected Conditions / Impacts</b>
The debug recovery sequence will not work when using the JTAG debug interface.
<b>Workaround</b>
Use the Serial Wire debug interface to implement the debug recovery sequence.
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.5 DCDC\_E204 – Potential Brownout when Enabling the DCDC from Off to Low Power Mode

<b>Description of Errata</b>
A brownout event (DVDD BOD) can occur if the DC-DC converter is enabled from off to Low Power mode.
<b>Affected Conditions / Impacts</b>
Firmware cannot set REGPWRSEL before setting EMU->DCDCCTRL.DCDCMODE=LOWPOWER.
<b>Workaround</b>
<p>To workaround this issue, firmware can:</p> <ol style="list-style-type: none"> <li>1. Set the EMU-&gt;DCDCPWRCTRL.REGPWRSEL bitfield to 1 to select DVDD.</li> <li>2. Set the EMU-&gt;DCDCCTRL.DCDCMODE bitfield to LOWNOISE.</li> <li>3. Wait for the DCDCLNRUNNING interrupt (or wait for the DCDCLNRUNNING bit to be set in the EMU_IF register).</li> <li>4. Set the EMU-&gt;DCDCCTRL.DCDCMODE bitfield to LOWPOWER.</li> </ol> <p>This workaround is included in v5.1.0 or later of the Gecko SDK.</p>
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.6 EMU\_E212 – Delay Required Between Successive Voltage Scaling Commands

<b>Description of Errata</b>
Issuing two successive unsuccessful voltage scaling commands may cause the device to brown out.
<b>Affected Conditions / Impacts</b>
Systems using voltage scaling should use one of the two workarounds to avoid an undesired brown out.
<b>Workaround</b>
<p>There are two workarounds for this problem:</p> <ol style="list-style-type: none"> <li>1. Always wait for the VSCALEDONE interrupt when voltage scaling up or down. This is a very easy solution to implement, but may cause a delay of up to 25 <math>\mu</math>s of delay when scaling up and up to 3 <math>\mu</math>s of delay when scaling down.</li> <li>2. For systems sensitive to delays or long voltage scaling up time: <ol style="list-style-type: none"> <li>a. When scaling down, wait for the VSCALEDONE (or VSCALEBUSY status) interrupt (up to 3 <math>\mu</math>s of delay).</li> <li>b. When scaling up, successive voltage scaling commands can be issued (no 25 <math>\mu</math>s delay required).</li> </ol> </li> </ol> <p>Successive voltage scaling commands should never be issued when scaling down.</p> <p>This workaround is included in v5.1.0 or later of the Gecko SDK.</p>
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.7 EMU\_E214 – Device Erase Cannot Occur if Voltage Scaling Level is Too Low

<b>Description of Errata</b>
The device erase logic does not check the Voltage Scale Level prior to attempting a device erase. If using Voltage Scale Level 0 (1 V), the device may not be able to erase the flash. This results in a potentially ununlockable device if operating at Voltage Scale Level 0 (1 V).
<b>Affected Conditions / Impacts</b>
It is possible that the flash is only partially erased when performing the operation at Voltage Scale Level 0 (1 V). If this results in the debug lock bit not clearing, a locked part doesn't unlock after the partial erasure (which it is intended to do), and the part remains locked. If subsequent erasures continue to fail, the part would remain locked.
<b>Workaround</b>
<p>The voltage should be set to Voltage Scale Level 2 (1.2 V) before executing the device erase.</p> <p>For systems that don't lock the debug interface, the user can follow the debug recovery procedure to halt the CPU before it has a chance to execute code in software to avoid the code scaling the voltage. The device erase can then be executed at Voltage Scale Level 2 (1.2 V) (the power-on default voltage of the device).</p> <p>For systems that do lock the debug interface, firmware can implement a mechanism whereby it can voltage scale or unlock debug access if its defined authentication method is passed.</p>
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.8 RMU\_E202 – External Debug Access Not Available After Watchdog or Lockup Full Reset

<b>Description of Errata</b>
When a watchdog reset or lockup reset is triggered in full-reset mode (RMU_CTRL.LOCKUPRMODE or RMU_CTRL.WDOGRMODE set to FULL or 4), a debugger will not be able to read AHB-AP or ARM core registers.
<b>Affected Conditions / Impacts</b>
Systems using the full reset mode for watchdog or lockup resets will see limited debugging capability after one of these resets triggers.
<b>Workaround</b>
Software should configure WDOGRMODE and LOCKUPRMODE to either LIMITED or EXTENDED mode if full debugger functionality is needed after a watchdog or lockup reset.
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.9 RMU\_E203 – AVDD Ramp Issue

<b>Description of Errata</b>
<p>The device may not properly start during power-on or restart when a voltage droop (brown out) occurs on AVDD. The failure is intermittent.</p> <p>For example configurations and waveforms that are more likely to result in this issue, see the following Knowledge Base article:</p> <p><a href="http://community.silabs.com/t5/32-bit-MCU-Knowledge-Base/RMU-E203-AVDD-Ramp-Issue/ta-p/197340">http://community.silabs.com/t5/32-bit-MCU-Knowledge-Base/RMU-E203-AVDD-Ramp-Issue/ta-p/197340</a></p> <p>To detect this failure state, place a GPIO toggle at the beginning of <code>main()</code> in the device firmware. When this failure occurs, the pin will not be toggling as expected, as the device is not executing any code.</p>
<b>Affected Conditions / Impacts</b>
Systems may intermittently see the device fail to start, reset, or respond. The current draw of the device in this state is ~100 µA and DECOUPLE will be fully powered (~1.2 V). The device will not execute any code in this state.
<b>Workaround</b>
<p>This issue can be resolved with a hardware workaround where an external circuit holds the reset pin low during power-on or brown out until AVDD reaches 1.8 V. For brown out, the reset pin must be configured to hard reset mode. This can be accomplished as part of the firmware image programmed to the device (lock bits area) or using the following code:</p> <pre>// Clears the CLW0 bit to enable Hard reset void enable_hardreset() {     uint32_t value;     uint32_t newvalue;     value = *(uint32_t *)0xFE041E8;     newvalue = value &amp; ~(1 &lt;&lt; 2);     MSC_WriteWord((uint32_t *)0xFE041E8, &amp;newvalue, 4); }</pre> <p>There is currently no software workaround for all potential failure mechanisms. The software workaround included in the Knowledge Base article will prevent failure in some scenarios. See the Knowledge Base article for more information:</p> <p><a href="http://community.silabs.com/t5/32-bit-MCU-Knowledge-Base/RMU-E203-AVDD-Ramp-Issue/ta-p/197340">http://community.silabs.com/t5/32-bit-MCU-Knowledge-Base/RMU-E203-AVDD-Ramp-Issue/ta-p/197340</a></p>
<b>Resolution</b>
This issue will be resolved in a future V2 module update.

## 2.10 RTCC\_E203 – Potential Stability Issue with RTCC Registers

<b>Description of Errata</b>
RTCC_LOCK and RTCC_POWERDOWN have the potential to be momentarily unstable under some PCLK, Low Energy Peripheral Clock, and APB write scenarios. This stability issue resolves in approximately 160 ns as the write completes with the assertion of the APB clock pulse.
<b>Affected Conditions / Impacts</b>
A write to RTCC_LOCK or RTCC_POWERDOWN may have unintended effects if the write is completed with the Low Energy Peripheral clock enabled (RTCC in the CMU_LFECLKEN0 register is set to 1).
<b>Workaround</b>
To avoid this stability issue, configure the RTCC_LOCK and RTCC_POWERDOWN registers with the Low Energy Peripheral clock disabled (RTCC in the CMU_LFECLKEN0 register is cleared to 0).
This workaround is included in v5.1.0 or later of the Gecko SDK.
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.11 RTCC\_E204 – Disabling the RTCC Backup RAM may Consume Extra Current

<b>Description of Errata</b>
Disabling the RTCC backup RAM may cause higher EM4H current draw due to the internal power structure of the backup RAM.
<b>Affected Conditions / Impacts</b>
Systems disabling the RTCC backup RAM may see additional current consumption.
<b>Workaround</b>
Firmware should keep the RTCC backup RAM retained. Leakage for the backup RAM is low (~3 nA typical), so the impact is slight.
This workaround is included in v5.1.0 or later of the Gecko SDK.
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.12 RTCC\_E205 – Wrap Event Can Be Missed

<b>Description of Errata</b>
The RTCC main counter can miss a CC1 wrap event (CCV1TOP bitfield in the RTCC_CTRL register set to 1) if one of the following registers are written in the same cycle as the wrap event: RTCC_CTRL, RTCC_CNT, RTCC_TIME, RTCC_DATE, RTCC_PRECNT, RTCC_IFC, RTCC_IFS, RTCC_CCx_CCV, RTCC_CCx_CTRL, RTCC_CCx_TIME, RTCC_CCx_DATE, RTCC_CMD, RTCC_RETx_REG.
<b>Affected Conditions / Impacts</b>
Systems using the CC1 wrap event feature may miss events if an affected register is written immediately before a wrap occurs.
<b>Workaround</b>
There are two workarounds to this issue:
<ul style="list-style-type: none"> <li>Do not use the CC1 wrap event feature (CCV1TOP in RTCC_CTRL should be cleared to 0).</li> <li>Alternatively, do not write to any of the affected registers when the counter is about to wrap. This means that firmware must check that RTCC_CNT is not close to RTCC_CC1_CCV before writing the register.</li> </ul>
<b>Resolution</b>
There is currently no resolution for this issue.



**2.13 USART\_E201 — USART DMA Transactions Fail with Slow Peripheral Clocks**

<b>Description of Errata</b>
USART DMA transactions will fail when the USART peripheral clock is slower than the DMA clock and IGNORESREQ is cleared to 0.
<b>Affected Conditions / Impacts</b>
Systems will not be able to use the DMA with a USART running from a slow clock when IGNORESREQ is cleared to 0.
<b>Workaround</b>
There is currently no workaround for this issue.
<b>Resolution</b>
There is currently no resolution for this issue.

**2.14 VDAC\_E201 — VDAC Output Drives All APORT Buses Simultaneously**

<b>Description of Errata</b>
<p>When VDACn_OPAX_OUT.APORTOUTEN is set, the VDACn/OPAx will drive all its connected APORT buses (BUSAY, BUSBY, BUSCY, and BUSDY) instead of only the APORT bus selected via APORTOUTSEL.</p> <p>The VDACn/OPAx APORT request signals do however correspond to the programmed APORTOUTSEL value, and therefore the APORT conflict registers (OPAxAPORTCONFLICT in VDACn_STATUS, OPAxAPORTCONFLICTIF in VDACn_IF, and VDACn_APORTCONFLICT) will not reflect potential conflicts on the erroneously driven APORT buses.</p> <p>If any other peripherals (or other VDAC channel or other OPAMP) are using either of the above APORT buses at the same time, this can lead to contention on the APORT bus and possible high current consumption.</p>
<b>Affected Conditions / Impacts</b>
Systems attempting to use multiple APORT buses and the VDAC may see contention.
<b>Workaround</b>
<p>If none of the other APORT clients (e.g. ADC, ACMP, OPAX input muxes, etc.) use BUSAY, BUSBY, BUSCY, and BUSDY, then no problem exists and the potential simultaneous driving of these buses by VDACn/OPAx can be ignored.</p> <p>Alternatively, the VDACn/OPAx can be configured to use direct connections of its main or alternative outputs to certain pins, thereby bypassing the APORT. Direct output connections can be enabled by programming MAINOUTEN=1 and/or ALTOUTEN=1 (while keeping APORTOUTEN=0) in the VDACn_OPAX_OUT register. The device data sheet lists the available main output and alternative output connections to pins per VDAC output or OPAMP.</p>
<b>Resolution</b>
There is currently no resolution for this issue.

## 2.15 VDAC\_E202 — PRS Outputs Not Generated when Interrupt Flag is Set

<b>Description of Errata</b>
<p>The conversion done (CD) PRS outputs from the DAC are tied to the interrupt flags. As long as the interrupt flag is set, no PRS output will be generated.</p> <p>When the first conversion done (CD) event occurs, the VDAC will set the interrupt flag and generate one PRS pulse. As long as the interrupt flag is set, any new conversion done events will not generate a new PRS pulse. After software clears the flag, the next conversion done event will generate a PRS pulse. Clearing the interrupt flag itself will not generate a pulse. Any CD event that occurs while the flag is set will be ignored.</p>
<b>Affected Conditions / Impacts</b>
<p>Systems attempting to use the DAC PRS outputs should ensure the interrupt flags are cleared.</p>
<b>Workaround</b>
<p>Firmware should clear the conversion done flag immediately after entering the interrupt service routine. This will allow the next conversion done event to generate a PRS pulse.</p>
<b>Resolution</b>
<p>There is currently no resolution for this issue.</p>

### 3. Errata History

This section contains the errata history for MGM12P devices.

For errata on latest revision, please 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 Errata History Summary

This table lists all resolved errata for the MGM12P.

**Table 3.1. Errata History Status Summary**

Errata #	Designator	Title/Problem	Workaround Exists	Affected Revision	Resolution
There are no errata in the errata history for this device.					

## 4. Revision History

### 4.1 Revision 0.1

June 12th, 2017

Initial release.

Silicon Labs

# Simplicity Studio™4



## 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](http://www.silabs.com/IoT)



**SW/HW**  
[www.silabs.com/simplicity](http://www.silabs.com/simplicity)



**Quality**  
[www.silabs.com/quality](http://www.silabs.com/quality)



**Support and Community**  
[community.silabs.com](http://community.silabs.com)

### Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice and limitation to product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Silicon Labs shall have no liability for the consequences of use of the information supplied herein. This document does not imply or express copyright licenses granted hereunder to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any Life Support System without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons.

### Trademark Information

Silicon Laboratories Inc.®, Silicon Laboratories®, Silicon Labs®, SiLabs® and the Silicon Labs logo®, Bluegiga®, Bluegiga Logo®, Clockbuilder®, CMEMS®, DSPLL®, EFM®, EFM32®, EFR®, Ember®, Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Ember®, EZLink®, EZRadio®, EZRadioPRO®, Gecko®, ISOModem®, Micrium, Precision32®, ProSLIC®, Simplicity Studio®, SiPHY®, Telegesis, the Telegesis Logo®, USBXpress®, Zentri and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. All other products or brand names mentioned herein are trademarks of their respective holders.



**SILICON LABS**

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

<http://www.silabs.com>