Apollo3 Blue MCU

Errata List

Doc. ID: SE-A3_2p00

Revision 2.0

Jul 2019

Table of Content

Silicon Errata for the Apollo3 Blue MCU (AMA3B1KK-xxx)

1. Introduction

This document is a compilation of detailed Silicon Errata for the Apollo3 Blue MCU.

2. Document Revision History

Rev #	Date	Description
1.0p	Oct 2018	Document initial release - preliminary, internal-only release
1.0	Apr 2019	Document initial public release
2.0	Jul 2019	ERR019 workaround updated; added ERR021

Table 1: Document Revision History

3. Errata Summary List

Below is a list of the errata described in this document. The reference number for each erratum is listed along with its description and link to the page where detailed information can be found.

Erratum Number, Title and Page	Affected Silicon Revisions	Resolution Status	Work-around
“ERR001: BLE: No transmissions on channel 0” on page 7	All revisions prior to B0	Fixed on B0	None
“ERR002: BLE Buck: Zerodetect brownout not cleared with buck_comp1 de-assertion” on page 8	All revisions	No fix scheduled	Software
“ERR003: CLKGEN: Register initializations are not retained from INFO space” on page 9	All revisions	No fix scheduled	Software
“ERR004: CTIMER: CTIMER read data may be incorrect” on page 10	All revisions	No fix scheduled	Software
“ERR005: Flash: CPU Flash accesses can get corrupted at burst disable” on page 11	All revisions	No fix scheduled	Software
“ERR006: Flash: Timing violation when switching from LPM mode 1 to 2” on page 12	All revisions	No fix scheduled	Software
“ERR007: Info0: No access to Info0 top half if PROTLOCK is open” on page 13	All revisions	No fix scheduled	Software
“ERR008: IOM: SPI read operation failure” on page 14	All revisions	No fix scheduled	Software
“ERR009: IOM: The FIFO Threshold (THR) interrupt could be asserted incorrectly in IOM” on page 15	All revisions	No fix scheduled	Software
“ERR010: IOM: Repeat Command feature is non-functional” on page 16	All revisions	No fix scheduled	Software, Functionality Deprecated
“ERR011: IOM: Auto power-off clock enable turns off too early” on page 18	All revisions	No fix scheduled	Software
“ERR012: MSPI: CONT functionality is broken for MSPI clock divider > 2” on page 20	All revisions	No fix scheduled	Software
“ERR013: MSPI: Multiple issues related to XIP and low-power mode transitions” on page 21	All revisions	No fix scheduled	Software
“ERR014: STIMER: Compare interrupt is not generated as expected” on page 22	All revisions	No fix scheduled	Software
“ERR015: IOM: I2C transactions without offset bytes may fail” on page 24	All revisions prior to B0	Fixed on B0	Software
“ERR016: BLE: Bluetooth® security vulnerability” on page 25	All revisions	No fix scheduled	Software

Table 2: Errata Summary

Erratum Number, Title and Page	Affected Silicon Revisions	Resolution Status	Work-around
“ERR017: BLE: Spurious out-of-band emission at around 800 MHz and 1600 MHz” on page 26	All revisions prior to B0	Fixed on B0	None
“ERR018: XTAL glitch causes HFADJ errors, incorrect frequency and failures” on page 27	All revisions prior to B0	Fixed on B0	Non-use of functionality
“ERR019: Core: SIMOBUCK locks up under certain conditions” on page 28	All revisions prior to B0	Fixed on B0	Software
“ERR020: BLE: Core update for elliptic-curve Die-Hellman (ECDH) vulnerability” on page 29	All revisions prior to B0	Fixed on B0	Software
“ERR021: GPIO: Possible higher than expected IO pad current” on page 30	Revision B0	No fix scheduled	None

Table 2: Errata Summary

4. Detailed Silicon Errata

This section gives detailed information about each erratum. Information covered for each erratum includes the following:

- **Erratum Reference Number and Title** – Lists reference number and title of the erratum
- **Description** – Provides a detailed description of the erratum
- **Affected Silicon Revisions** – Specifies the silicon revisions on which the erratum exists
- **Application Impact** – Describes the impact of the erratum on a user application
- **Workarounds** – Proposes software or hardware workarounds to minimize or eliminate the risk of the erratum occurring
- **Erratum Resolution Status** – Specifies which silicon revision, if any, that the erratum was initially fixed
- **AmbiqSuite Workaround Status** – Specifies whether the erratum has been worked around in the AmbiqSuite software

4.1 ERR001: BLE: No transmissions on channel 0

4.1.1 Description

Channel 0 (2404 MHz) of the BLE is non-functional and does not communicate as expected.

4.1.2 Affected Silicon Revisions

This silicon erratum applies to all existing revisions of Apollo3 Blue prior to B0. Fixed on B0.

4.1.3 Application Impact

BLE applications experiences a slight degradation in performance because of this issue. Due to the use of channel hopping as per the Bluetooth protocol, any packet transmission expected but failing to take place on channel 0 will hop to another channel to retransmit immediately after missing the packet on channel 0.

4.1.4 Workarounds

This issue cannot and does not need to be worked around in software, as the fix is built into the Bluetooth Baseband protocol.

4.1.5 Erratum Resolution Status

This erratum is fixed in B0.

4.1.6 AmbiqSuite Workaround Status

No workaround is needed in AmbiqSuite.

4.2 ERR002: BLE Buck: Zerodetect brownout not cleared with buck_comp1 de-assertion

4.2.1 Description

The blebuck_low brownout indicator does not get cleared when buck_comp1 referenced in the CLKGEN_BLEBUCKTONADJ_ZEROLENDETECTTRIM field) is de-asserted in some cases. In case of a BLE brownout event with BLE brownout reset enabled, the RSTGEN issues a POI and this gets cleared with it. However, if this is used with just interrupt, then this does not get cleared even though buck_comp1 de-asserts.

4.2.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue.

4.2.3 Application Impact

When used in interrupt-only mode, the BLE Buck brownout interrupt is not cleared.

4.2.4 Workarounds

Clear the BLE Buck brownout interrupt manually by disabling and enabling the ZEROLENDETECTEN field as part of the ISR.

```
AM_BFW(CLKGEN,BLEBUCKTONADJ,ZEROLENDETECTEN,0);
AM_BFW(CLKGEN,BLEBUCKTONADJ,ZEROLENDETECTEN,1);
```

4.2.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.2.6 AmbiqSuite Workaround Status

AmbiqSuite does not have or need a workaround for this issue - workaround code above needs to be in a user's ISR.

4.3 ERR003: CLKGEN: Register initializations are not retained from INFO space

4.3.1 Description

The following registers within the CLKGEN module will not be pre-loaded at power-on-reset with initialization values programmed into the INFO space. These registers must be updated with the target values after reset and boot have completed. The initial value of the registers will be 0.

The affected registers fields are:

- REG_CLK_GEN_CALXT (address 0x40004000) - All non-reserved fields affected
- REG_CLK_GEN_CALRC (address 0x40004004) - All non-reserved fields affected
- REG_CLK_GEN_BLEBUCKTONADJ (address 0x4000403c) – All non-reserved fields affected
- REG_CLK_GEN_OCTRL (address 0x4000400c) - OSEL field only affected.

These fields should be updated by software with the target values for the application after the MCU has booted. The values will be maintained through a POI, but will be reset on a POA event.

The fields contain user-updated information, and no field is used to hold device-unique data.

4.3.2 Affected Silicon Revisions

This silicon erratum applies to all existing revisions of Apollo3 Blue.

4.3.3 Application Impact

Low impact, as the features enabled with these settings are only used after boot, and in limited circumstances.

4.3.4 Workarounds

The MCU software must program these values if needed after the boot process.

4.3.5 Erratum Resolution Status

Currently there are no plans to fix this issue on the Apollo3 device.

4.3.6 AmbiqSuite Workaround Status

A workaround is implemented in the AmbiqSuite SDK,starting with release 2.0. The BLEBUCKTONADJ is initialized in the BLE HAL function, am_hal_ble_power_control(). The other affected fields are handled on an as-needed basis, e.g., examples that use the RTC always set OSEL accordingly and do not assume a setting for it.

4.4 ERR004: CTIMER: CTIMER read data may be incorrect

4.4.1 Description

The CTIMER module updates the timer counters using a different clock domain than the CPU system bus interface. If a CPU read cycle occurs at the same time the CTIMER updates its internal count register, the return data may be erroneous, but the internal count data remains reliable and intact. In slower devices, the data from the CTIMERs does not make it to the read_data register in time and read errors result. The errors only occur if:

- Using an HFRC-based clock source.
- The CTIMER is clocked on the same clock cycle in which the read_data register is loaded.

4.4.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue.

4.4.3 Application Impact

An erroneous CTIMER value could be read.

4.4.4 Workarounds

Read the CTIMER three times. See the AmbiqSuite Workaround Status section below.

4.4.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.4.6 AmbiqSuite Workaround Status

AmbiqSuiteHAL release 2.0 adds reading the CTIMER three times, which is implemented in the HAL function am_hal_ctimer_read() in the SDK.

4.5 ERR005: Flash: CPU Flash accesses can get corrupted at burst disable

4.5.1 Description

It is possible to corrupt flash read accesses during burst mode transitions if the following conditions occur:

- Cache miss from the ARM (this is the fetch that fails).
- Transitioning out of burst mode when either a DMA from flash or an info access from flash occurs at the exact right timing usually with heavy flash DMA traffic and caches off.
- Another flash access occurs immediately after the ARM access (DMA or info read). In this scenario, the flash returns data to the ARM and expects it to capture data within two DMA clock cycles. However, the burst transition off gates HCLK for 2+ cycles, which can delay the capture point. Flash will not update the read data for 2 cycles, however, if another access comes in, the data can get updated before the ARM's clock resumes and allows it to capture the data.

4.5.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue.

4.5.3 Application Impact

Flash may not be read correctly during burst mode transitions.

4.5.4 Workarounds

If customer code doesn't perform DMA operations from flash, then there is no issue. If it does, then the simplest workaround is to not enable burst mode during the operation. Also, execute the AmbiqSuite HAL code that safely transitions out of burst mode. This ensures that no ARM flash accesses occur near the burst transition boundary.

4.5.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.5.6 AmbiqSuite Workaround Status

AmbiqSuite contains a function, am_hal_burst_mode_disable(), in the Burst Mode Hal that will safely write CLKGEN_FREQCTRL_BURSTREQ_DIS.

4.6 ERR006: Flash: Timing violation when switching from LPM mode 1 to 2

4.6.1 Description

When CACHECTRL_FLASHCFG_LPMMODE changes from 1 (Fast Standby mode) to 2 (Low Power mode) while FLASH accesses are active, it causes a timing violation when LPM is asserted.

4.6.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue.

4.6.3 Application Impact

LPM Mode may not transition reliably.

4.6.4 Workarounds

The workaround is to ensure FLASH is idle when switching between LPM modes. This includes ensuring that no DMA accesses of FLASH might occur during the mode transitioning. Any code modifying LPM Mode must *not* be executing in on-chip flash. See the AmbiqSuite workaround for an example implementation.

4.6.5 Erratum Resolution Status

Currently there are no plans to fix this issue on the Apollo3 Blue device.

4.6.6 AmbiqSuite Workaround Status

The HAL in AmbiqSuite has a function that will safely handle the update of LPMMODE, am_hal_cachectrl_control(AM_HAL_CACHECTRL_CONTROL_LPMMODE_SET). If this function is not used for updating of LPMMODE, the write and read back of the value must be done from code that is not executing from Apollo3 on-chip flash.

4.7 ERR007: Info0: No access to Info0 top half if PROTLOCK is open

4.7.1 Description

Currently, customer infospace access restrictions are tied to the MCUCTRL_BOOTLOADER_SBLOCK (Secure boot lock) bit instead of PROTLOCK (Flash protection lock) bit, requiring the use of CUSTOMER_KEY in customer bootloader to access the keys. This restricts the flexibility for the customer bootloader (e.g. per part unique customer key).

4.7.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue.

4.7.3 Application Impact

Could severely restrict the flexibility, e.g. per part unique customer key, of the customer bootloader.

4.7.4 Workarounds

Since SBL has access to the customer key before SBLOCK is asserted, SBL reads the customer key and sets in the key registers before passing control to customer bootloader. This changes the default behavior of this flag as visible to the customer, i.e., it is unlocked.

The customer bootloader must write some value to the key registers before moving to the firmware to protect the infospace, in addition to asserting the PROTLOCK.

SBL implements the workaround only if the device is configured to have a secondary bootloader (indicated by means of INFO0->SECURITY.PLONEXIT). The additional requirement for the customer bootloader is applicable only in that case.

4.7.5 Erratum Resolution Status

Currently there are no plans to fix this issue on the Apollo3 Blue device.

4.7.6 AmbiqSuite Workaround Status

AmbiqSuite release 2.0 has updated SBL, however the second step by the customer bootloader described in the Workarounds section still needs to be implemented by the user.

4.8 ERR008: IOM: SPI read operation failure

4.8.1 Description

When a SPI device requires a 1-byte command followed by a 24-byte offset address, or 4 bytes total, multiple operations are required as the 4-byte "offset" cannot be handled in a single operation. If a transfer is initiated with the first command byte sent as a read-operation with a 1-byte offset value, a 0-length transfer size, and using CONT, the IOM will attempt to read data until the FIFO is filled, as the transfer count is incorrectly decremented from 0.

4.8.2 Affected Silicon Revisions

This silicon erratum applies to all existing revisions of Apollo3 Blue, and the described behavior occurs in all IOM instances using either communication channel.

4.8.3 Application Impact

Impact is erroneous SPI operation under a specific transfer configuration.

4.8.4 Workarounds

The workaround is to replace a 0-byte read with a 0-byte write.

4.8.5 Erratum Resolution Status

Currently there are no plans to resolve the erratum, as it can be easily worked around in software.

4.8.6 AmbiqSuite Workaround Status

A workaround is implemented in the AmbiqSuite SDK,starting with release 2.0 to never do a 0-length read to avoid the effect of this erratum.

4.9 ERR009: IOM: The FIFO Threshold (THR) interrupt could be asserted incorrectly in IOM

4.9.1 Description

The THR interrupt may get asserted inadvertently and unpredictably as a result of intermediate values of incoming signals before settling, causing the THR bit in the INTSTAT register to be set. This event triggers an interrupt if the THR bit is set in the INTEN register. This issue only affects the threshold interrupts when used for software-based FIFO servicing. DMA transfers, which also depend on thresholds, are not impacted.

4.9.2 Affected Silicon Revisions

This silicon erratum applies to all existing revisions of Apollo3 Blue, and the described behavior occurs in all IOM instances using either communication channel.

4.9.3 Application Impact

This erratum could cause random, erroneous FIFO threshold interrupts to occur, and if the FIFO status is not verified in software, it could result in untimely reading or sending of data.

4.9.4 Workarounds

Either do not use the THR interrupt in real-time operations or, if it is used, check the actual FIFO level (FIFOSIZ) before sending or reading data.

4.9.5 Erratum Resolution Status

Currently there are no plans to resolve the erratum, as it can be easily worked around in software.

4.9.6 AmbiqSuite Workaround Status

No workaround for this issue is currently implemented in the SDK HAL, and the THR interrupt is not handled in the am_hal_iom_interrupt_service() routine. However, the am_hal_iom_configure() function does set both FIFOWTHR and FIFORTHR to default values for DMA purposes.

4.10 ERR010: IOM: Repeat Command feature is non-functional

4.10.1 Description

Write with Repeat:

- For a transfer length (TLNGTH) of 1 or 2, CMDRPT does not complete the transaction with a repeat count of 1. Since software writes only one word (size 2 or 4, respectively), the whole word gets consumed by IOM in the first transaction itself and gets stuck waiting for more data for the second transaction.
- For a TLNGTH of 3, CMDRPT completes the transaction, but the data is not right. IOM consumes FIFO data in word multiples for each transaction and software has to write 6 bytes, or 2 words. Even though the IOM consumes a whole word for the first transaction, there is still one more word in the FIFO for the second transaction to consume, and therefore it does not get stuck. However, the data sent out on the line is wrong.

Read with Repeat:

- For a repeat count of 1, two CMDCMP interrupts are raised - one for each transaction.
- For other values of repeat count, only one CMDCMP interrupt is raised as expected.

NOTE: CMDRPT functionality has been deprecated starting with silicon version B0. See Workarounds section for details.

4.10.2 Affected Silicon Revisions

This silicon erratum applies to all existing revisions of Apollo3 Blue, and the described behavior occurs in all IOM instances.

4.10.3 Application Impact

Since CMDRPT does not work reliably, software must explicitly execute the same transaction the required number of times without the benefit of having the hardware *automatically* repeat the command the specified number of times.

4.10.4 Workarounds

Do not attempt to use the Repeat Command feature. The CMDRPT function has been deprecated, and reference to it and the CMDRPT registers has been removed from the Apollo3 Blue datasheet, version 0.9 and later. Starting with silicon version B0, these registers, one for each IOM instance, are repurposed for totally different functionality (DCX Control Register) and incorrectly or inadvertently writing to these registers may have adverse results.

This erratum serves as a notification for customers, who might have based their development on earlier versions of the datasheet and CMDRPT functionality, not to attempt to use these IOM registers for CMDRPT functionality.

4.10.5 Erratum Resolution Status

Currently there are no plans to resolve the erratum, as it can be easily worked around in software by not using the CMDRPT function.

4.10.6 AmbiqSuite Workaround Status

AmbiqSuite does not use the CMDRPT feature of the IOM, and therefore does not encounter this issue.

4.11 ERR011: IOM: Auto power-off clock enable turns off too early

4.11.1 Description

The IOM auto power-off feature does not work in some cases because the clock is disabled before the operation can be done, as the clock disable does not take into account synchronization that may be needed to fully complete the power down request.

4.11.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue, and the described behavior occurs in all IOM instances when using either communication channel.

4.11.3 Application Impact

The IOM may not get powered off as expected thereby causing excessive current, especially in deep sleep mode.

4.11.4 Workarounds

This feature requires the use of the Command Queue (CQ) and DMA for transactions. For the last CQ DMA transaction (DMACFG write with DPWROFF=1), an additional CQ entry to enable the IO CLK should be done. This is done by setting bit 1 of the IOM_IOMDBG register.

4.11.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.11.6 AmbiqSuite Workaround Status

This feature is currently not used or exposed in the AmbiqSuite HAL because the state is not retained (or restored).

The AmbiqSuite HAL provides a usage module enforced by the IOM driver API. The normal sequence of driver operation is:

- Initialization & Configuration
  - am_hal_iom_initialize
  - am_hal_iom_power_ctrl(power-up)
  - am_hal_iom_configure
  - am_hal_iom_enable
- Operation
  - am_hal_iom_blocking_transfer, or
  - am_hal_iom_nonblocking_transfer & interrupt handling
- Shutdown
  - am_hal_iom_disable
  - am_hal_iom_power_ctrl(power-down)

- am_hal_iom_uninitialize

In addition, power can be saved during normal operation by making the following call:

- am_hal_iom_power_ctrl(power-down, save-state)

and then restore the state upon power-up as follows:

- am_hal_iom_power_ctrl(power-up, restore-state)

4.12 ERR012: MSPI: CONT functionality is broken for MSPI clock divider > 2

4.12.1 Description

The MSPI CONT function (ability to bridge two independent transfers while holding CE low) is broken for cases where MSPICFG_CLKDIV is not set to CLK24 (i.e., less than 24 MHz MSPI clock) and the second transaction is RX only. CONT is not used for flash operations. An additional RX capture phase before the first clock edge of the second transfer causes an extra bit to be captured.

4.12.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue.

4.12.3 Application Impact

MSPI operation may take slightly longer as bridging two transfers at an MSPI frequency of less than 24MHz cannot be done.

4.12.4 Workarounds

Do not use CONT when clocking the MSPI at less than 24MHz.

4.12.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.12.6 AmbiqSuite Workaround Status

AmbiqSuite will not use this functionality, as it is not a supported (or documented) function of the MSPI module.

4.13 ERR013: MSPI: Multiple issues related to XIP and low-power mode transitions

4.13.1 Description

Specific issues are:

- When MSPI is left powered on entering deepsleep, the core can't switch to LP mode.
- In the above mode, when power to the XIP Flash Controller is dropped, synchronization logic between Flash and MSPI may be in a state that results in the MSPI performing a fetch from the configured XIP device.
- If MSPI is powered down, SW execution must jump to internal Flash for the power-down sequence in order to re-initialize the XIP registers upon wakeup from deep sleep. A CQ sequence residing in flash can be used to minimize ARM overhead.

4.13.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue.

4.13.3 Application Impact

There is no negative impact when power to the XIP Flash Controller is dropped other than an unused MSPI read operation. However, this could be problematic if power to MSPI is too high for the LP power source (assuming the override is present to enable LP mode). Powering MSPI down and back up doesn't result in any negative effects. The sync logic can launch a false transaction, but since MSPI is powered off, XIP is disabled. However, the chip enable to device 0 will go low while a false XIP operation is in progress. No data or clock pins toggle since the MSPI is not configured.

4.13.4 Workarounds

Use the code in the MSPI HAL (am_hal_mspi_power_control) to save and restore the MSPI state during MSPI power down. The HAL must reside in internal Flash and not subject to XIP. Thus, if the XIP program calls these routines, the condition will be satisfied.

4.13.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.13.6 AmbiqSuite Workaround Status

In the AmbiqSuite SDK release 2.0, the MSPI HAL (am_hal_mspi_power_control) has added functions to save and restore the MSPI state during power down.

4.14 ERR014: STIMER: Compare interrupt is not generated as expected

4.14.1 Description

The STIMER compare register may capture invalid data as the add operation is in the process of computing the new value. Thus there is a race condition between the STIMER clock and the interface clock.

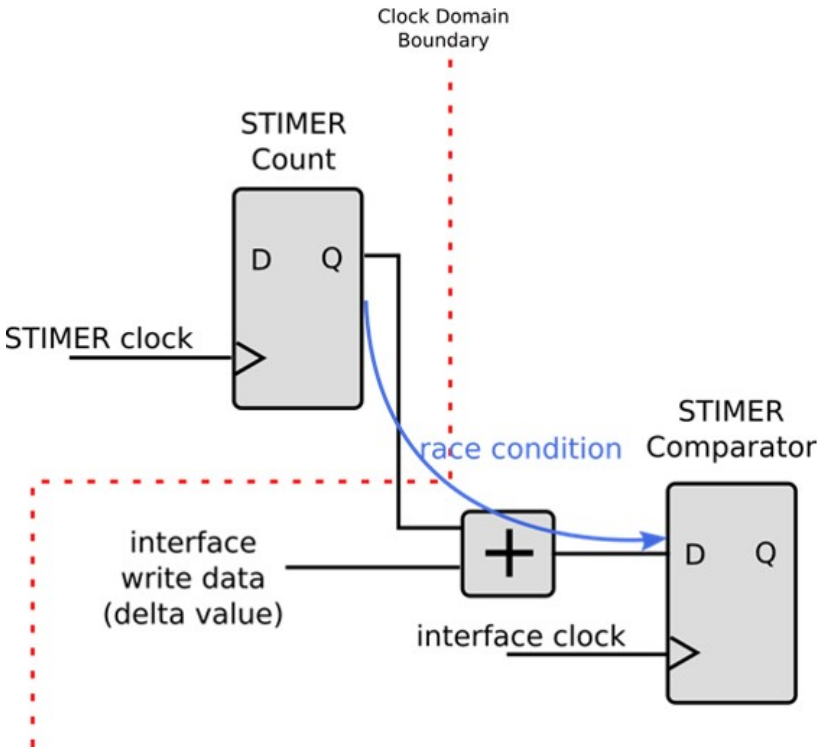

Figure 1. STIMER Compare Register Update Race Condition

4.14.2 Affected Silicon Revisions

This silicon erratum applies to all existing revisions of Apollo3 Blue.

4.14.3 Application Impact

Mistimed STIMER interrupts could occur as a result of invalid compare values being written to the compare register.

4.14.4 Workarounds

A workaround is to create a critical section in code to set the STIMER compare value without the possibility that invalid data will get captured for the STIMER clock value. The general program flow for the critical section would be as follows:

Start Critical Section

Disable STIMER Compare

For N=4 Attempts:

Compute the expected compare value by reading the register and adding the delta value

Compute an expected maximum compare value to account for latency

Set the STIMER Compare Register (SCMPRx) with the delta value

Read back the STIMER Compare Register

Check if returned value is within expected latency

If it is, then exit. Else, retry

Enable STIMER Compare

End Critical Section

4.14.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.14.6 AmbiqSuite Workaround Status

AmbiqSuite implements the critical section safeguard described above to prevent this erratum from occurring.

4.15 ERR015: IOM: I2C transactions without offset bytes may fail

4.15.1 Description

The byte count is not correctly stopped when an I2C write transaction is started with the command only, running in immediate mode, and no data is sent with the command. There is a stretch event when the address is transferred and, in this case, the byte count of the transferred data does not get correctly decremented and may result in a hang if only the exact amount of data is delivered to the FIFO.

4.15.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue prior to B0.

4.15.3 Application Impact

I2C may hang if only the exact amount of data is delivered to the I2C FIFO and an additional transfer is attempted.

4.15.4 Workarounds

Always provide the write data to the FIFO before issuing the command to avoid this issue.

4.15.5 Erratum Resolution Status

This erratum is fixed in B0.

4.15.6 AmbiqSuite Workaround Status

AmbiqSuite SDK release 2.1.0 updated the HAL blocking call to make sure the FIFO is written before the CMD is initiated. For non-blocking calls, the command queue is used and the issue does not exist.

4.16 ERR016: BLE: Bluetooth® security vulnerability

4.16.1 Description

There are Bluetooth pairing vulnerabilities reported on the NIST website as:

- CVE ID: CVE-2018-5383
- CVSS: 7.1 CVSS 3.0/AV:A/AC:H/PR:N/UI:R/S:U/C:H/I:H/A:H

Two of the issues affecting Bluetooth Low Energy (BLE) on Apollo3 Blue are the following:

- Repeated attempts which is a host feature only. Only host code is required
- Random Key Attack

4.16.2 Affected Silicon Revisions

This silicon erratum applies to all existing revisions of Apollo3 Blue.

4.16.3 Application Impact

Pairing vulnerability.

4.16.4 Workarounds

A fix from Cordio host stack is provided by re-generating a private key after each pairing whether it is successful or not.

4.16.5 Erratum Resolution Status

This erratum is intended to be fixed in a future chip revision.

4.16.6 AmbiqSuite Workaround Status

In AmbiqSuite SDK version 2.0.0, a workaround for this issue was implemented by forcing the regeneration of a new private key in the SMP layer.

4.17 ERR017: BLE: Spurious out-of-band emission at around 800 MHz and 1600 MHz

4.17.1 Description

When the Apollo3's BLE is transmitting during advertising and other normal activities, it produces unexpected out-of-band spurious emission at around 800 MHz and 1600 MHz. This abnormal out-of-band emission is caused by timing issues of the power amplifier (PA) input.

4.17.2 Affected Silicon Revisions

This silicon erratum applies to all revisions of Apollo3 Blue prior to B0.

4.17.3 Application Impact

The spurious emission could cause poor BLE performance and lower signal integrity when advertising, connecting or transmitting data.

4.17.4 Workarounds

None.

4.17.5 Erratum Resolution Status

This erratum is fixed in revision B0.

4.17.6 AmbiqSuite Workaround Status

The Ambiqsuite SDK does not implement a workaround for this issue.

4.18 ERR018: XTAL glitch causes HFADJ errors, incorrect frequency and failures

4.18.1 Description

Occasionally there is excessive activity in the analog module, potentially causing the frequency of the HFRC to go out of the supported range and cause unusual faults to occur. Glitching on XTAL is most likely to occur with high IOM activity, buck mode & high VDDA supply, where coupling is believed to be a contributor.

4.18.2 Affected Silicon Revisions

This silicon erratum applies to all versions of Apollo3 Blue prior to B0.

4.18.3 Application Impact

Incorrect HFRC frequency could cause application timing issues and cause device to run out of specification.

4.18.4 Workarounds

Do not use HFADJ. Also, since all SBL prior to SBLv3 (applicable to all pre-B0 versions of silicon) may enable HFADJ under certain conditions, it is recommended to disable HFADJ as part of initialization.

4.18.5 Erratum Resolution Status

Starting in revision B0, a fix makes the XTAL clock more robust to noise, where a bias change increases current to the XTAL comparator and routing differences balance capacitance seen by XI and XO.

4.18.6 AmbiqSuite Workaround Status

The AmbiqSuite SDK does not implicitly enable HFADJ.

4.19 ERR019: Core: SIMOBUCK locks up under certain conditions

4.19.1 Description

Under certain conditions and setup, the SIMOBUCK mode of voltage regulation may lock up and fail to regulate both the VDDC and VDDF power domains. The voltage domains will degrade to low voltage and cause a brown out event to occur.

4.19.2 Affected Silicon Revisions

This silicon erratum applies to all versions of Apollo3 Blue prior to B0.

4.19.3 Application Impact

VDDC and VDDF voltage domains may drop to a voltage which causes a brown out event to occur, which will result in a reset of the device.

The power impact to the system design is nominally around 20-30 uA in Sleep Mode

4.19.4 Workarounds

The following operational modes will prevent the issue from occurring:

- Run the internal voltage regulator in LDO mode, or...
- aintain a power domain active at all times within the device. This will prevent the SIMOBUCK module from going to low power mode. The power domain which is the least impact on power is the PDM power domain, which can be enabled by setting the PWRPDM bit in the PWRCTRL_DEVPWREN register.

4.19.5 Erratum Resolution Status

Starting with revision B0, this issue has been resolved.

4.19.6 AmbiqSuite Workaround Status

A fix is implemented in AmbiqSuite SDK 2.2 for the above workaround, applied only when operating on an A1 device.

4.20 ERR020: BLE: Core update for elliptic-curve Die-Hellman (ECDH) vulnerability

4.20.1 Description

There is a security vulnerability related to a man-in-the-middle attack on the pairing procedures as described by the CERT Division in the Vulnerability note entitled “Bluetooth implementations may not sufficiently validate elliptic curve parameters during Diffie-Hellman key exchange”.

4.20.2 Affected Silicon Revisions

This silicon erratum applies to all versions of Apollo3 Blue prior to B0.

4.20.3 Application Impact

Security vulnerabilities.

4.20.4 Workarounds

Use ECC math library instead of BLE controller to validate public keys from remote device, and reject pairing in security manager protocol layer if they are invalid public keys.

4.20.5 Erratum Resolution Status

Starting with revision B0, this vulnerability issue is resolved by a new version of the CEVA IP incorporated in the design.

4.20.6 AmbiqSuite Workaround Status

In AmbiqSuite SDK version 2.0.0, a workaround for this issue was implemented by integrating an open source ECC library (named uECC) into the SDK to validate the public key from the remote device. First, if the validation fails, the ongoing pairing will be rejected immediately in the SMP layer without further engaging with BLE controller for subsequent ECDH key generations.

4.21 ERR021: GPIO: Possible higher than expected IO pad current

4.21.1 Description

On silicon revision B0, there can be higher than expected IO pad current (from the VDDH supply) when using either a drive strength setting of 2 or 3. The pad current could be up to twice as high compared to silicon revision A1 when the supply voltage is greater than 2V.

4.21.2 Affected Silicon Revisions

This silicon erratum applies to revision B0 of Apollo3 Blue.

4.21.3 Application Impact

Higher than expected/necessary system power draw.

4.21.4 Workarounds

For supply voltages greater than 2V, a drive strength setting of 0 or 1 is recommended (GPIO_PADREGx_PADnSTRNG = 0 or 1, GPIO_ALTPADCFGx_PADn_DS1 = 0).

4.21.5 Erratum Resolution Status

Currently there are no plans to fix this issue on Apollo3 Blue.

4.21.6 AmbiqSuite Workaround Status

AmbiqSuite does not have or need a workaround for this issue - workaround code above needs to be in a user's GPIO configuration code.

Contact Information

Address	Ambiq Micro, Inc. 6500 River Place Blvd. Building 7, Suite 200 Austin, TX 78730-1156
Phone	+1 (512) 879-2850
Website	https://www.ambiqmicro.com/
General Information	info@ambiqmicro.com
Sales	sales@ambiqmicro.com
Technical Support	https://support.ambiqmicro.com

Legal Information and Disclaimers