

## ERRATA LIST

### **Apollo3 Blue Plus SoC**

Ultra-low Power Apollo SoC Family

Doc. ID: SE-A3P-6p0

Doc. Revision: 6.0, December 2024



## Legal Information and Disclaimers

AMBIQ MICRO INTENDS FOR THE CONTENT CONTAINED IN THE DOCUMENT TO BE ACCURATE AND RELIABLE. THIS CONTENT MAY, HOWEVER, CONTAIN TECHNICAL INACCURACIES, TYPOGRAPHICAL ERRORS OR OTHER MISTAKES. AMBIQ MICRO MAY MAKE CORRECTIONS OR OTHER CHANGES TO THIS CONTENT AT ANY TIME. AMBIQ MICRO AND ITS SUPPLIERS RESERVE THE RIGHT TO MAKE CORRECTIONS, MODIFICATIONS, ENHANCEMENTS, IMPROVEMENTS AND OTHER CHANGES TO ITS PRODUCTS, PROGRAMS AND SERVICES AT ANY TIME OR TO DISCONTINUE ANY PRODUCTS, PROGRAMS, OR SERVICES WITHOUT NOTICE.

THE CONTENT IN THIS DOCUMENT IS PROVIDED "AS IS". AMBIQ MICRO AND ITS RESPECTIVE SUPPLIERS MAKE NO REPRESENTATIONS ABOUT THE SUITABILITY OF THIS CONTENT FOR ANY PURPOSE AND DISCLAIM ALL WARRANTIES AND CONDITIONS WITH REGARD TO THIS CONTENT, INCLUDING BUT NOT LIMITED TO, ALL IMPLIED WARRANTIES AND CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHT.

AMBIQ MICRO DOES NOT WARRANT OR REPRESENT THAT ANY LICENSE, EITHER EXPRESS OR IMPLIED, IS GRANTED UNDER ANY PATENT RIGHT, COPYRIGHT, MASK WORK RIGHT, OR OTHER INTELLECTUAL PROPERTY RIGHT OF AMBIQ MICRO COVERING OR RELATING TO THIS CONTENT OR ANY COMBINATION, MACHINE, OR PROCESS TO WHICH THIS CONTENT RELATE OR WITH WHICH THIS CONTENT MAY BE USED.

USE OF THE INFORMATION IN THIS DOCUMENT MAY REQUIRE A LICENSE FROM A THIRD PARTY UNDER THE PATENTS OR OTHER INTELLECTUAL PROPERTY OF THAT THIRD PARTY, OR A LICENSE FROM AMBIQ MICRO UNDER THE PATENTS OR OTHER INTELLECTUAL PROPERTY OF AMBIQ MICRO.

INFORMATION IN THIS DOCUMENT IS PROVIDED SOLELY TO ENABLE SYSTEM AND SOFTWARE IMPLEMENTERS TO USE AMBIQ MICRO PRODUCTS. THERE ARE NO EXPRESS OR IMPLIED COPYRIGHT LICENSES GRANTED HEREUNDER TO DESIGN OR FABRICATE ANY INTEGRATED CIRCUITS OR INTEGRATED CIRCUITS BASED ON THE INFORMATION IN THIS DOCUMENT. AMBIQ MICRO RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN. AMBIQ MICRO MAKES NO WARRANTY, REPRESENTATION OR GUARANTEE REGARDING THE SUITABILITY OF ITS PRODUCTS FOR ANY PARTICULAR PURPOSE, NOR DOES AMBIQ MICRO ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT, AND SPECIFICALLY DISCLAIMS ANY AND ALL LIABILITY, INCLUDING WITHOUT LIMITATION CONSEQUENTIAL OR INCIDENTAL DAMAGES. "TYPICAL" PARAMETERS WHICH MAY BE PROVIDED IN AMBIQ MICRO DATA SHEETS AND/OR SPECIFICATIONS CAN AND DO VARY IN DIFFERENT APPLICATIONS AND ACTUAL PERFORMANCE MAY VARY OVER TIME. ALL OPERATING PARAMETERS, INCLUDING "TYPICALS" MUST BE VALIDATED FOR EACH CUSTOMER APPLICATION BY CUSTOMER'S TECHNICAL EXPERTS. AMBIQ MICRO DOES NOT CONVEY ANY LICENSE UNDER NEITHER ITS PATENT RIGHTS NOR THE RIGHTS OF OTHERS. AMBIQ MICRO PRODUCTS ARE NOT DESIGNED, INTENDED, OR AUTHORIZED FOR USE AS COMPONENTS IN SYSTEMS INTENDED FOR SURGICAL IMPLANT INTO THE BODY, OR OTHER APPLICATIONS INTENDED TO SUPPORT OR SUSTAIN LIFE, OR FOR ANY OTHER APPLICATION IN WHICH THE FAILURE OF THE AMBIQ MICRO PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR. SHOULD BUYER PURCHASE OR USE AMBIQ MICRO PRODUCTS FOR ANY SUCH UNINTENDED OR UNAUTHORIZED APPLICATION, BUYER SHALL INDEMNIFY AND HOLD AMBIQ MICRO AND ITS OFFICERS, EMPLOYEES, SUBSIDIARIES, AFFILIATES, AND DISTRIBUTORS HARMLESS AGAINST ALL CLAIMS, COSTS, DAMAGES, AND EXPENSES, AND REASONABLE ATTORNEY FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PERSONAL INJURY OR DEATH ASSOCIATED WITH SUCH UNINTENDED OR UNAUTHORIZED USE, EVEN IF SUCH CLAIM ALLEGES THAT AMBIQ MICRO WAS NEGLIGENT REGARDING THE DESIGN OR MANUFACTURE OF THE PART.

## Table of Content

Introduction .....	4
Document Revision History.....	4
Errata Summary List .....	5
Detailed Silicon Errata .....	7
ERR002: BLE Buck: Zerodetect brownout not cleared with buck_comp1 de-assertion .....	8
ERR003: CLKGEN: Register initializations are not retained from INFO space .....	9
ERR004: CTIMER: CTIMER read data may be incorrect.....	10
ERR005: Flash: CPU flash accesses can get corrupted at burst disable.....	11
ERR006: Flash: Timing violation when switching from LPM mode 1 to 2 .....	12
ERR007: Info0: No access to Info0 top half if PROTLOCK is open .....	13
ERR008: IOM: SPI read operation failure .....	14
ERR009: IOM: The FIFO Threshold (THR) interrupt could be asserted incorrectly in IOM....	15
ERR010: IOM: Repeat Command feature is non-functional.....	16
ERR011: IOM: Auto power-off clock enable turns off too early .....	18
ERR012: MSPI: CONT functionality is broken for MSPI clock divider > 2.....	20
ERR013: MSPI: Multiple issues related to XIP and low-power mode transitions .....	21
ERR014: STIMER: Compare interrupt is not generated as expected .....	22
ERR016: BLE: Bluetooth® security vulnerability .....	24
ERR022: MSPI and IOM operations at 48 MHz have limitations .....	25
ERR023: MSPI0 XIP address access failure.....	27
ERR024: Bytes 0-7 at 1 MB offset of flash not read protectable .....	28
ERR025: MSPI: Mixed Mode D4 does not work as expected .....	29
ERR026: STIMER: Glitch on XTAL clock input may cause missed compare .....	30
ERR027: I2S: Module has been deprecated and use of interface is not recommended .....	31
ERR028: VDDC/VDDF brownout during deepsleep/wake transition in buck mode .....	32
ERR029: Flash: Bit flip due to VDDH/VDDF power-up race condition .....	33
ERR030: MSPI: Incorrect data reads at 48 MHz may occur .....	35
ERR032: GPIO: Possible glitch on GPIO outputs upon initial power up .....	36
ERR033: GPIO51 does not transition high using lowest drive strength on some devices .....	37

# Silicon Errata for the Apollo3 Blue Plus SoC (AMA3B2KK-KBR)

## 1. Introduction

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

## 2. Document Revision History

Rev #	Date	Description
1.0	Dec 2019	Document initial public release
2.0	Jan 2020	ERR022 renamed and updated; ERR023 added
3.0	Mar 2020	ERR024 added
4.0	Dec 2021	ERR025 - ERR028 added
5.0	May 2022	ERR029 - ERR030 added
5.1	Jul 2022	ERR029 updated
5.2	Aug 2022	ERR029 workaround updated
5.3	Sep 2022	ERR029 workaround expanded
6.0	Dec 2024	ERR032, ERR033 added

**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
“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 PROTECT-LOCK 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
“ERR016: BLE: Bluetooth® security vulnerability” on page 24	All revisions	No fix scheduled	Software
“ERR022: MSPI and IOM operations at 48 MHz have limitations” on page 25	All revisions up to rev. C0	Partially fixed	Non-use of functionality
“ERR023: MSPI0 XIP address access failure” on page 27	All revisions	No fix scheduled	None-use of functionality

**Table 2: Errata Summary**

Erratum Number, Title and Page	Affected Silicon Revisions	Resolution Status	Work-around
“ERR024: Bytes 0-7 at 1 MB offset of flash not read protectable” on page 28	All revisions	No fix scheduled	None-use of functionality
“ERR025: MSPI: Mixed Mode D4 does not work as expected” on page 29	All revisions	No fix scheduled	None-use of functionality
“ERR026: STIMER: Glitch on XTAL clock input may cause missed compare” on page 30	All revisions	No fix scheduled	Software
“ERR027: I2S: Module has been deprecated and use of interface is not recommended” on page 31	All revisions	No fix scheduled	Software, Functionality Deprecated
“ERR028: VDDC/VDDF brownout during deepsleep/wake transition in buck mode” on page 32	All revisions	No fix scheduled	Software
“ERR029: Flash: Bit flip due to VDDH/VDDF power-up race condition” on page 33	All revisions	No fix scheduled	Hardware
“ERR030: MSPI: Incorrect data reads at 48 MHz may occur” on page 35	All revisions	No fix scheduled	Software
“ERR032: GPIO: Possible glitch on GPIO outputs upon initial power up” on page 36	All revisions	No fix scheduled	None
“ERR033: GPIO51 does not transition high using lowest drive strength on some devices” on page 37	All revisions	No fix scheduled	Software

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 ERR002: BLE Buck: Zerodetect brownout not cleared with buck\_comp1 de-assertion**

### **4.1.1 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.

### **4.1.2 Affected Silicon Revisions**

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

### **4.1.3 Application Impact**

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

### **4.1.4 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);
```

### **4.1.5 Erratum Resolution Status**

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

### **4.1.6 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.



## 4.2 ERR003: CLKGEN: Register initializations are not retained from INFO space

### 4.2.1 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.

### 4.2.2 Affected Silicon Revisions

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

### 4.2.3 Application Impact

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

### 4.2.4 Workarounds

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

### 4.2.5 Erratum Resolution Status

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

### 4.2.6 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.

## 4.3 ERR004: CTIMER: CTIMER read data may be incorrect

### 4.3.1 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.

### 4.3.2 Affected Silicon Revisions

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

### 4.3.3 Application Impact

An erroneous CTIMER value could be read.

### 4.3.4 Workarounds

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

### 4.3.5 Erratum Resolution Status

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

### 4.3.6 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.

## 4.4 ERR005: Flash: CPU flash accesses can get corrupted at burst disable

### 4.4.1 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.

### 4.4.2 Affected Silicon Revisions

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

### 4.4.3 Application Impact

Flash may not be read correctly during burst mode transitions.

### 4.4.4 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.

### 4.4.5 Erratum Resolution Status

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

### 4.4.6 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`.

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

### 4.5.1 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.

### 4.5.2 Affected Silicon Revisions

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

### 4.5.3 Application Impact

LPM Mode may not transition reliably.

### 4.5.4 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.

### 4.5.5 Erratum Resolution Status

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

### 4.5.6 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.

## **4.6 ERR007: Info0: No access to Info0 top half if PROTLOCK is open**

### **4.6.1 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).

### **4.6.2 Affected Silicon Revisions**

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

### **4.6.3 Application Impact**

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

### **4.6.4 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.

### **4.6.5 Erratum Resolution Status**

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

### **4.6.6 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.

## 4.7 ERR008: IOM: SPI read operation failure

### 4.7.1 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.

### 4.7.2 Affected Silicon Revisions

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

### 4.7.3 Application Impact

Impact is erroneous SPI operation under a specific transfer configuration.

### 4.7.4 Workarounds

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

### 4.7.5 Erratum Resolution Status

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

### 4.7.6 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.

## **4.8 ERR009: IOM: The FIFO Threshold (THR) interrupt could be asserted incorrectly in IOM**

### **4.8.1 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.

### **4.8.2 Affected Silicon Revisions**

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

### **4.8.3 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.

### **4.8.4 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.

### **4.8.5 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**

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 FIFORTH to default values for DMA purposes.

## 4.9 ERR010: IOM: Repeat Command feature is non-functional

### 4.9.1 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.**

### 4.9.2 Affected Silicon Revisions

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

### 4.9.3 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.

### 4.9.4 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 Plus 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.

### 4.9.5 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.



#### **4.9.6 *AmbiqSuite Workaround Status***

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

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

### 4.10.1 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.

### 4.10.2 Affected Silicon Revisions

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

### 4.10.3 Application Impact

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

### 4.10.4 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.

### 4.10.5 Erratum Resolution Status

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

### 4.10.6 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)`

## **4.11 ERR012: MSPI: CONT functionality is broken for MSPI clock divider > 2**

### **4.11.1 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.

### **4.11.2 Affected Silicon Revisions**

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

### **4.11.3 Application Impact**

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

### **4.11.4 Workarounds**

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

### **4.11.5 Erratum Resolution Status**

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

### **4.11.6 AmbiqSuite Workaround Status**

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

## **4.12 ERR013: MSPI: Multiple issues related to XIP and low-power mode transitions**

### **4.12.1 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.

### **4.12.2 Affected Silicon Revisions**

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

### **4.12.3 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.

### **4.12.4 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.

### **4.12.5 Erratum Resolution Status**

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

### **4.12.6 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.

## 4.13 ERR014: STIMER: Compare interrupt is not generated as expected

### 4.13.1 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.

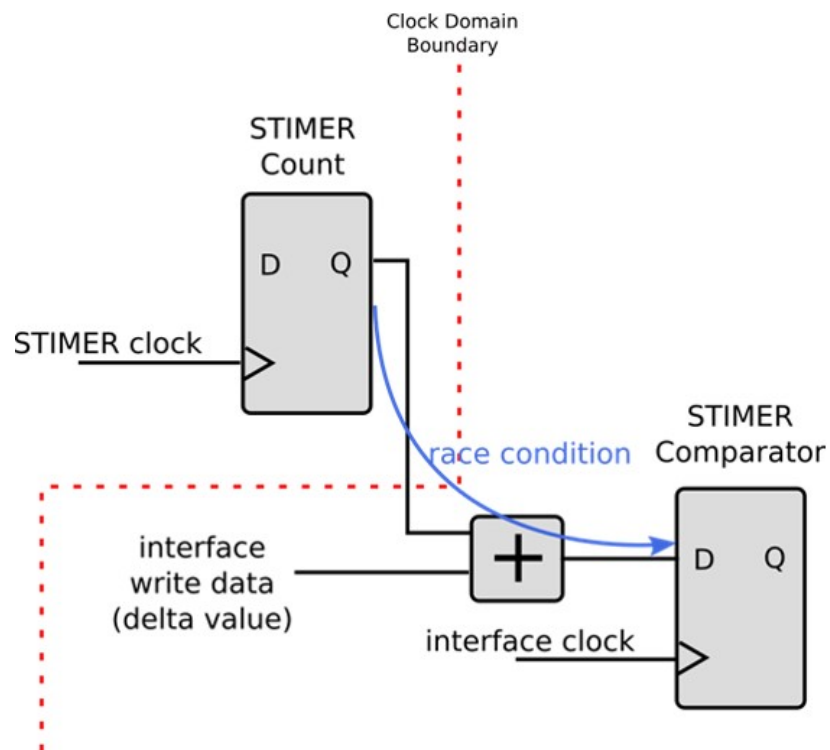


Figure 1. STIMER Compare Register Update Race Condition

### 4.13.2 Affected Silicon Revisions

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

### 4.13.3 Application Impact

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

#### **4.13.4 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

#### **4.13.5 Erratum Resolution Status**

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

#### **4.13.6 AmbiqSuite Workaround Status**

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

## **4.14 ERR016: BLE: Bluetooth® security vulnerability**

### **4.14.1 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 Plus are the following:

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

### **4.14.2 Affected Silicon Revisions**

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

### **4.14.3 Application Impact**

Pairing vulnerability.

### **4.14.4 Workarounds**

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

### **4.14.5 Erratum Resolution Status**

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

### **4.14.6 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.



## 4.15 ERR022: MSPI and IOM operations at 48 MHz have limitations

NOTE: This erratum has been renamed and updated from Apollo3 Blue Plus Errata List, version 1.0. It was formerly called “MSPI2 and IOM1 are limited to 24 MHz reads”, but its status has changed and the former name is no longer applicable.

### 4.15.1 Description

In version 1.0 of the Apollo3 Blue Plus Errata List, this erratum was listed with the following issues and limitations:

- Multi-bit SPI module 2 (MSPI2) is limited to 24 MHz clock speed during read operations. If strictly performing write operations, then MSPI2 can be clocked at 48 MHz.
- I2C/SPI Master module 1 (IOM1) is limited to 24 MHz clock speed during read operations. If strictly performing write operations, then IOM1 can be clocked at 48 MHz.

In version 2.0, after further validation this erratum has been updated to specify that:

- Multi-bit SPI module 2 (MSPI2), as well as MSPI0 and MSPI1, can reliably perform reads and writes at 48 MHz clock speed in quad, dual and single-bit modes. However, the capacitance loading on any MSPI signal (CLK and data lines) is limited to no higher than 30 pF. In addition, the minimum MISO input data setup time is 3 ns.
- Octal mode is supported on MSPI1, but the speed is limited to 24 MHz for both read and write operation with the same setup time restrictions.
- I2C/SPI Master module 1 (IOM1) is no longer limited to 24 MHz clock speed during read operations. All IOMs can reliably perform both reads and writes at 48 MHz clock speed. However, the capacitance loading on any IOM signal (CLK, MOSI, MISO) at that maximum clock frequency is limited to no higher than 25 pF. In addition, the minimum MISO input data setup time is 4 ns.

### 4.15.2 Affected Silicon Revisions

This silicon erratum applies to revision C0 of Apollo3 Blue Plus.

### 4.15.3 Application Impact

For single-bit, dual and quad MSPI modes and for IOM operations at 48 MHz, the loading and setup time limitations mentioned above must be met for proper operation. Care should be taken to ensure that the board design and any peripheral connected and operating at 48 MHz meet these requirements.

For MSPI operation at 48 MHz, the CLKDIV field of the MSPIn Control Register (MSPIn\_MSPICTRL) should be changed from the default value of 0x2 (CLK24) for a clocking frequency of 24 MHz, to 0x1 (CLK48) for a clocking frequency of 48 MHz

Octal mode with MSPI1 must be run at no higher than 24 MHz.

### 4.15.4 Workarounds

If board design or peripheral timing cannot meet the above-mentioned limitations, then MSPI or IOM operation should be done at a lower clock rate than 48 MHz.

#### **4.15.5 Erratum Resolution Status**

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

#### **4.15.6 AmbiqSuite Workaround Status**

The AmbiqSuite SDK ensures that the required setup time of 3 ns for MSPI operation at 48MHz, as well as any timing constraints at other frequencies when operating at 3V, are met by leaving the MSPIn DDR Configuration Bits Register at its default value.

AmbiqSuite leaves it to the user to specify a suitable clock rate for any IOM or MSPI operation, and that all loading and timing requirements are met for reliable operation.

## 4.16 ERR023: MSPI0 XIP address access failure

### 4.16.1 Description

A bus fault error is triggered when trying to access certain addresses in instance 0 (MSPI0) of the 32 MB Multi-bit SPI execute-in-place (XIP) address range. The internal flash copy protection settings in the FLASHRPROT0-FLASHRPROT3 registers in the MCU Control module, or indirectly through the COPYPROTECT0 - COPYPROTECT3 registers in the customer INFO space (INFO0), are incorrectly applied to the 32 MB MSPI0 XIP space such that the assigned protection is mirrored in each 2MB (0x0020\_0000) boundary. Copy protection is automatically implemented for the first 48K (first three chunks up to 0x0000\_BFFF) of the 2MB flash, so consequently the first 48K of *each* 2MB sector of the 32MB MSPI0 XIP space cannot be used for XIP. It can be used though for memory-mapped data reads and writes (XIPMM).

Flash write protection settings in the MCU Control module's FLASHWPROT0-FLASHWPROT3 registers [or INFO0 (WRITEPROTECT0 - WRITEPROTECT3)] do not impact or apply to the MSPI0 space.

This issue does not impact MSPI1 or MSPI2 XIP.

### 4.16.2 Affected Silicon Revisions

This silicon erratum applies to revision C0 of Apollo3 Blue Plus.

### 4.16.3 Application Impact

An application is not able to execute code from certain locations of the MSPI0's XIP region. Those regions include the first 48K of each 2MB sector of the 32MB MSPI0 XIP region, as well as those chunks corresponding to any address ranges that the user sets as protected in the flash copy protection registers.

### 4.16.4 Workarounds

To work around this issue, It is recommended that the user not use MSPI0 for XIP operations. He can use MSPI0 for data fetches from memory-mapped locations in the XIP space. If the user wants or needs to use MSPI0 for XIP operations, then he should limit the amount of flash that is copy protected, as that same copy protection will be applied to each 2MB sector of the MSPI0 XIP space. Not only will applying copy protection reduce the total amount of available XIP space, it might also create non-contiguous execution areas which would present complications when linking these regions of code.

### 4.16.5 Erratum Resolution Status

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

### 4.16.6 AmbiqSuite Workaround Status

AmbiqSuite does not perform any error checking to prevent attempted code execution from copy-protected areas of the MSPI0 XIP space. It is up to the user to ensure that XIP operation in copy-protected XIP space is not attempted in the code.

## **4.17 ERR024: Bytes 0-7 at 1 MB offset of flash not read protectable**

### **4.17.1 Description**

Read and copy protection does not work for the first 2 words (8 bytes) of chunk 64. Chunk 64 is halfway through the 2 MB flash memory space on the Apollo3 Blue Plus MCU. Specifically, the bytes which cannot be read- and copy-protected are at addresses 0x00100000-0x00100007. Write protection is not affected by this erratum.

### **4.17.2 Affected Silicon Revisions**

This silicon erratum applies to all revisions of the Apollo3 Blue Plus MCU.

### **4.17.3 Application Impact**

The internal flash copy protection settings in the FLASHRPROT0-FLASHRPROT3 registers in the MCU Control module, or indirectly through the COPYPROTECT0 - COPYPROTECT3 registers in the customer INFO space (INFO0), are incorrectly applied to the first 8 bytes at the 1MB offset of flash such that the assigned protection is not enforced for these 8 bytes.

There is no impact to code execution due to this “non-protection” issue.

### **4.17.4 Workarounds**

If it of concern that these 8 bytes of flash cannot be protected from attempted copying or reading, then the user can arrange his link map so that non-protection of these 8 bytes does not cause undesired exposure of data at this location.

### **4.17.5 Erratum Resolution Status**

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

### **4.17.6 AmbiqSuite Workaround Status**

AmbiqSuite does not perform any error checking to prevent attempted code execution from copy-protected areas of the internal flash space, nor does it throw an error when an attempt is made to protect any memory that is not able to be read or copy protected.

## **4.18 ERR025: MSPI: Mixed Mode D4 does not work as expected**

### **4.18.1 Description**

MSPI mixed mode D4 (FLASH\_XIPMIXED = D4), or 1-1-4 mode, where data operations (only) occur in quad data mode, corrupts bits 7-4 of the first byte due to MSPI lines D1-D3 incorrectly staying low.

### **4.18.2 Affected Silicon Revisions**

This silicon erratum applies to all revisions of the Apollo3 Blue Plus MCU.

### **4.18.3 Application Impact**

The value read for the first byte of data in this mode will be incorrect.

### **4.18.4 Workarounds**

Use MSPI mixed mode AD4 (1-4-4 mode), as this mode works correctly.

### **4.18.5 Erratum Resolution Status**

There are no plans to fix this issue on the Apollo3 Blue Plus MCU.

### **4.18.6 AmbiqSuite Workaround Status**

AmbiqSuite supports MSPI mixed mode AD4 (1-4-4).

## **4.19 ERR026: STIMER: Glitch on XTAL clock input may cause missed compare**

### **4.19.1 Description**

The STIMER is sensitive to glitches on the incoming clock source when clocked with the XTAL. This susceptibility sometimes results in double pulses (counter incrementing twice) which could cause the comparator to not trigger on the first pulse resulting in a missed compare and a timer overflow.

### **4.19.2 Affected Silicon Revisions**

This silicon erratum applies to all revisions of the Apollo3 Blue and Apollo3 Blue Plus MCUs.

### **4.19.3 Application Impact**

A missed STIMER compare action may cause indeterminable and undesirable behavior in the application.

### **4.19.4 Workarounds**

A second compare whose value is one count higher than the initial compare value should be established. This second compare will ensure that if the first compare is missed because of a glitch-induced double pulse, the second compare will work and intended compare processing can take place.

### **4.19.5 Erratum Resolution Status**

There are no plans to fix this issue on Apollo3 Blue or Apollo3 Blue Plus MCUs.

### **4.19.6 AmbiqSuite Workaround Status**

AmbiqSuite supports the ability to set multiple STIMER compares. It is the responsibility of the application to take the appropriate actions in each compare interrupt's ISR.

## **4.20 ERR027: I2S: Module has been deprecated and use of interface is not recommended**

### **4.20.1 Description**

The I2S module on the Apollo3 Blue and the Apollo3 Blue Plus MCUs has been deprecated. Proper and expected operation the I2S interface cannot be guaranteed and its use is not recommended.

### **4.20.2 Affected Silicon Revisions**

This issue is applicable to all silicon revisions of Apollo3 Blue and Apollo3 Blue Plus MCUs.

### **4.20.3 Application Impact**

Any applications using the I2S interface may encounter feature limitations or various unexpected operation.

### **4.20.4 Workarounds**

Other serial interfaces are available on the MCU to share PDM data with an external host.

### **4.20.5 Erratum Resolution Status**

There are no plans to fix this issue on Apollo3 Blue or Apollo3 Blue Plus MCUs.

### **4.20.6 AmbiqSuite Workaround Status**

The AmbiqSuite SDK (still) supports the pass-through of PDM data to the I2S slave.

## 4.21 ERR028: VDDC/VDDF brownout during deepsleep/wake transition in buck mode

### 4.21.1 Description

Both VDDC and VDDF drop to below brownout level when transitioning from deepsleep mode to active mode in buck mode. Brownouts may occur with either 1.8 V or 3.3 V VDD. In order to prevent SIMOBUCK brownouts, VDDC and VDDF voltage levels in deep sleep need to be 50-60 mV higher than in active mode before the next exit from deep sleep cycle. Therefore, brownouts are more likely to occur with very short incursions into deep sleep mode, as the VDDC/VDDF are not rising adequately during these short deep sleep cycles.

### 4.21.2 Affected Silicon Revisions

This issue is applicable to all silicon revisions of Apollo3 Blue and Apollo3 Blue Plus MCUs.

### 4.21.3 Application Impact

Unexpected brownouts may occur, especially when short incursions are made into deepsleep mode.

### 4.21.4 Workarounds

To work around this potential brownout issue, the following steps should be taken:

- Before entering deep sleep, set the VDDC and VDDF SIMOBUCK active trim levels to 60 mV above the normal active voltage levels. This can be done by increasing the SIMOBUCK core active trim code by 24 codes and the simobuck mem active trim code by 9 codes.
- Wait for rails to charge up by 60 mV. This should take only a few microseconds since the SIMOBUCK is running in active mode.
- After the rails charge up by 60 mV, set the SIMOBUCK core and SIMOBUCK mem active trim codes back to the original values and immediately enter deepsleep mode. This should prevent any voltage drop on the rail since the chip will enter deepsleep with the SIMOBUCK low power voltages at the same level (trimmed 60 mV above the active).

When the chip exits deepsleep, the VDDC and VDDF voltage levels will be 60 mV above the active mode targets.

### 4.21.5 Erratum Resolution Status

There are no plans to fix this issue on Apollo3 Blue or Apollo3 Blue Plus MCUs.

### 4.21.6 AmbiqSuite Workaround Status

The above workaround has been implemented in the `am_hal_sysctrl_sleep` function in the HAL of the AmbiqSuite SDK.



## 4.22 ERR029: Flash: Bit flip due to VDDH/VDDF power-up race condition

### 4.22.1 Description

The SIMO Buck output, VDDF, which supplies internal flash, may not ramp up to a proper voltage level upon slow VDDH ramp-up. A section of the flash memory/logic is powered up before VDDF reaches the required voltage to maintain the correct state of the flash memory logic, which could cause bit cells in the flash memory to become erased.

### 4.22.2 Affected Silicon Revisions

This issue is applicable to all silicon revisions of Apollo3 Blue Plus SoC.

### 4.22.3 Application Impact

Bit flips in flash could cause improper execution of program code or corruption of stored data.

### 4.22.4 Workarounds

Following are the workaround options for this power-up timing issue.

1. If the VDDH ramp-up time, i.e. the time it takes for VDDH to transition from 0 V to final supply voltage, is less than 180  $\mu$ s or greater than 100 ms, then there is no danger of a race condition that may cause flash corruption so no hardware workaround is needed.
2. If the VDDH ramp-up time is greater than 180  $\mu$ s but less than 10 ms, then the use of a bootstrap capacitor between VDDF and VDDP/VDDH is recommended to prevent the issue as follows:
  - A. Bootstrap VDDF to the VDDP/VDDH voltage supply by inserting a capacitor between the supply and the SIMO Buck / LDO output pin, VDDF. The capacitor between the supply and VDDF should have a value according to the VDDP/VDDH voltage range in the table below. This ensures that VDDF is properly coupled to VDDP/VDDH to provide the correct VDDF ramp rate and voltage level, which ensures the correct state of the flash memory logic during power up.

VDDP/VDDH Range	VDDP/VDDF Capacitance Value
1.75 V - 2.2 V	2.2 $\mu$ F
2.1 V - 2.75 V	1.8 $\mu$ F
2.72 V - 3.6 V	1.5 $\mu$ F

- B. With the bootstrap capacitor installed, VDDF should initially be boosted to 850 mV to 1.2 V, ultimately settling to the final VDDF target of around 860 mV. It should also be ensured, based on the bootstrap capacitor value used and the VDDH power up characteristics, that when VDDF is ramping up, the VDDF voltage overshoot does not exceed 1.4 V (max initial ramp voltage of 1.2 V + 0.2 V), and that it does not stay between 1.2 V and 1.4 V for more than 10 ms. After adding the bootstrap capacitor, the VDDF ramp rate should be observed and the capacitor value should be adjusted, if necessary, to meet these requirements.

3. If the VDDH ramp-up time is between 10 ms and 100 ms, then the ramp-up rate should be slowed to 100 ms or longer, or reduced to 10 ms or less. If the latter is done, then use the bootstrap capacitor workaround and adhere to the requirements described in (2) above.

#### **4.22.5 Erratum Resolution Status**

There are no plans to fix this issue on Apollo3 Blue or Apollo3 Blue Plus MCUs.

#### **4.22.6 AmbiqSuite Workaround Status**

There is no software workaround in the AmbiqSuite SDK.

## **4.23 ERR030: MSPI: Incorrect data reads at 48 MHz may occur**

### **4.23.1 Description**

The first byte of data in response to a command from the MCU may be read incorrectly at an MSPI's clock rate of 48 MHz. After the MCU sends a command, the data line is locked low for a clock cycle and then is released. If the receiving device does not observe or incur any turnaround time and returns a response to a command immediately as might be the case when reading a device's QPID, the first byte of the response may not be received correctly.

### **4.23.2 Affected Silicon Revisions**

This issue is applicable to all MSPI instances using any data width on all silicon revisions of Apollo3 Blue Plus SoC.

### **4.23.3 Application Impact**

Some data reads of certain devices at 48 MHz may not be reliable and may require multiple reads.

### **4.23.4 Workarounds**

Recommended workaround is to re-send the command and/or read the response multiple times, or reduce the MSPI clock to 24 MHz or slower.

### **4.23.5 Erratum Resolution Status**

There are no plans to fix this issue on Apollo3 Blue Plus SoC.

### **4.23.6 AmbiqSuite Workaround Status**

The above workaround options are available as standard read functions in the HAL of the AmbiqSuite SDK.

## **4.24 ERR032: GPIO: Possible glitch on GPIO outputs upon initial power up**

### **4.24.1 Description**

There is a possibility of experiencing a glitch on GPIO outputs due to the initial state of an internal level shifter being bi-stable during power up when those outputs are temporarily pulled up to VDD\_MCU.

### **4.24.2 Affected Silicon Revisions**

This issue is applicable to affected GPIO on all silicon revisions of Apollo3 Blue Plus SoC.

### **4.24.3 Application Impact**

The effect of this issue is dependent upon the sensitivity to a glitch on the circuit/peripheral that is connected to the GPIO. It is likely to not have a detrimental effect.

### **4.24.4 Workarounds**

There currently is no workaround for this issue.

### **4.24.5 Erratum Resolution Status**

There are no plans to fix this issue on Apollo3 Blue Plus SoC.

### **4.24.6 AmbiqSuite Workaround Status**

The AmbiqSuite SDK does not provide a software workaround for this issue.

## **4.25 ERR033: GPIO51 does not transition high using lowest drive strength on some devices**

### **4.25.1 Description**

On a small percentage of devices, GPIO51 does not output a high level when configured to do so if the drive strength is set to the lowest level, nominally 2 ma (PADREGM\_PAD51STRNG = LOW, ALTPADCFGM\_PAD51\_DS1 = 0) while driving no load.

### **4.25.2 Affected Silicon Revisions**

This issue is applicable to all silicon revisions of Apollo3 Blue Plus SoC.

### **4.25.3 Application Impact**

The impact of a GPIO not outputting a high level when programmed and expected to do so is design-dependent and the severity of such a failure should be analyzed by the designer. Using the next highest drive strength as a workaround does not cause any significant increase in overall power consumption.

### **4.25.4 Workarounds**

The workaround for this issue is to use the next highest drive strength, nominally 4 ma drive, by setting PADREGM\_PAD51STRNG = HIGH. Alternatively, use another GPIO.

### **4.25.5 Erratum Resolution Status**

There are no plans to fix this issue on Apollo3 Blue Plus SoC.

### **4.25.6 AmbiqSuite Workaround Status**

The AmbiqSuite SDK supports the setting of all GPIO drive strengths.

**BLANK PAGE**



©2024 All rights reserved.

Ambiq Micro, Inc.

6500 River Place Boulevard, Building 7,

Suite 200, Austin, TX 78730-1156

[www.ambiq.com/](http://www.ambiq.com/)

[sales@ambiqmicro.com](mailto:sales@ambiqmicro.com)

<https://support.ambiqmicro.com>

+1 (512) 879-2850

SE-A3P-6p0

Version 6.0

December 2024