

## Errata

# AM263x Sitara™ 微控制器

## 器件版本 1.0A , 1.1



### 摘要

本文档介绍了功能规格的已知例外情况 ( 公告 )。本文档也包含了使用说明。在使用说明中介绍了器件行为可能与假定或记录的行为不匹配的情况。这可能包括影响器件性能或功能正确性的行为。

### 内容

1 使用说明和公告模型.....	2
2 器件使用说明.....	4
3 器件公告.....	5
4 修订历史记录.....	25

## 1 使用说明和公告模型

表 1-1 列出了所有使用说明和适用的器件修订版本。表 1-2 列出了所有公告、受影响的模块以及适用的器件修订版本。

表 1-1. 使用说明汇总表

模块	说明	受影响的器件修订版本	
		AM263x	
		1.0A	1.1
时钟	i2324 — GCM 和 GCD 状态信号之间不存在同步器	是	是
QSPI	i2364 — QSPI：在存储器映射模式下不支持访问超过 8MB 的地址	是	是
安全	i2508 — 安全系统中 RC OSC 的使用	是	是
VDDA	i2348 — VDDA1V8 静态功率泄漏	是	否

表 1-2. 公告汇总表

模块	说明	受影响的器件修订版本	
		AM263x	
		1.0A	1.1
ADC	i2346 — 在奇数通道和偶数通道之间切换时 ADC 结果出现误差	是	否
ADC	i2347 — ADC 的 VREF 电流消耗在上电时为随机	是	否
ADC	i2349 — 断电时 ADC VrefHi 负载增加	是	否
AES	i2428 — DTHE 中的 AES 在 GCM 加密结束时为 data_in 生成额外的 dma 请求	是	是
总线安全	i2393 — BUS_SAFETY_ERR 寄存器中未记录检测到的故障的粒度错误状态	是	是
时钟	i2488 — 时钟：用于精确 50-50 占空比时钟的 PLL 配置	是	是
CONTROLSS	i2352 — CONTROLSS-SDFM：动态更改阈值设置 (LLT、HLT)、滤波器类型或 COSR 设置将触发虚假比较器事件	是	是
CONTROLSS	i2353 — CONTROLSS-SDFM：动态更改数据滤波器设置 (例如滤波器类型或 DOSR) 将触发虚假数据响应事件	是	是
CONTROLSS	i2354 — CONTROLSS-SDFM：在三个 SD 调制器时钟周期内对 SDCPARMx 寄存器位字段 CEVT1SEL、CEVT2SEL 和 HZEN 进行两次背对背写入可能会损坏 SDFM 状态机，从而导致虚假比较器事件	是	是
CONTROLSS	i2355 — CONTROLSS-ADC：DMA 读取过时结果	是	是
CONTROLSS	i2356 — CONTROLSS-ADC：如果未设置 INTxCONT (继续中断模式)，中断可能会停止	是	是
CONTROLSS	i2357 — CONTROLSS-ePWM：如果跳闸在消隐窗口结束时保持活动状态，则可能会发生 ePWM 干扰	是	是
CONTROLSS	i2358 — CONTROLSS-ePWM：在消隐开始后的前 3 个周期内，消隐窗口不会过滤跳闸事件	是	是
CONTROLSS	i2359 — CONTROLSS-CMPSS：当 DACSOURCE 设为 0 或重新配置为 1 时，预分频器计数器的行为与规范不同	是	是
CONTROLSS	i2405 — CONTROLSS：竞争条件 OUTPUT_XBAR 和 PWM_XBAR 导致事件丢失	是	是
CPSW	i2345 — CPSW：如果 CPDMA 提取跨越存储器组的数据包，则会发生以太网数据包损坏	是	是
CPSW	i2401 — CPSW：主机时间戳导致 CPSW 端口锁定	是	是
CPSW	i2402 — CPSW：以太网到主机校验和卸载不起作用	是	是
CPSW	i2438 — CPSW - 带 VLAN 添加/删除功能的主机转以太网校验和生成	是	是
CPSW	i2439 — CPSW：主机到以太网时间戳精度问题	是	是
CRC	i2386 — CRC：不支持 CRC 8 位数据宽度以及 CRC8-SAE-J1850 和 CRC8-H2F 在 CAN 模块中的潜在用途	是	是
DCC	i2395 — DCC 模块频率比较可报告错误结果	是	是
闪存	i2503 — 在闪存引导模式下，无法从闪存的冗余引导位置进行引导	是	是
ICSS	i2433 — ICSS：读取 LSW 时，读取 64 位 IEP 计时器没有锁定 MSW 逻辑	是	是
GPMC	i2313 — GPMC：NAND 和 FPGA/FIFO 的低于 32 位读取问题	是	是
LIN	i2500 — 使用 0xF0 作为唤醒密钥时 LIN 模块无法唤醒	是	是
M4 ROM	i2403 — M4 ROM：在 HSSE 器件上不支持 SBL 冗余启动映像特性	否	是

表 1-2. 公告汇总表 (续)

模块	说明	受影响的器件修订版本	
		AM263x	
		1.0A	1.1
MBOX	i2404 — MBOX：邮箱寄存器中的竞争条件导致事件丢失	是	是
McSPI	i2350 — 在“ABSYNCR”模式下使用 EDMA 的 McSPI 数据传输在 32 位传输后停止	是	是
MDIO	i2329 — MDIO 接口损坏 ( CPSW 和 PRU-ICSS )	是	否
PBIST	i2374 — 如果 R5SS_CORE_CLK 的时钟频率与 R5FSS_CLK_SELECTED 频率不同，则 PBIST 失败	是	是
PBIST	i2502 — PBIST 存储器组与目标存储器之间的映射不正确	是	是
RAM	i2499 — 突发读取期间发生单错误检测时，返回给主机的数据不正确	是	是
RAM SEC	i2427 — RAM SEC 可能会导致虚假的 RAM 写入，从而导致 L2 和 MBOX 存储器损坏	是	是
SDFM	i2375 — 如果比较器事件仍处于活动状态并且正在选择数字滤波器路径 ( 使用 SDCOMPxCTL.CEVTxDIGFILSEL )，则不会再次设置 SDFM 模块事件标志 (SDIFLG.FLTx_FLG_CEVTx)	是	是
SOC CONTROL	i2392 — 存储器初始化捕获寄存器中的竞争条件导致事件丢失	是	是
SOC CONTROL	i2394 — 中断和错误聚合器捕获寄存器中的竞争条件导致事件丢失	是	是
UART	i2310 — USART：错误地触发超时中断	是	是
UART	i2311 — USART：虚假 DMA 中断	是	是

## 2 器件使用说明

### **i2324** *GCM 和 GCD 状态信号之间不存在同步器*

---

**详细信息：** GCM 和 GCD 之间没有同步器，因此时钟配置寄存器的读取可能会暂时不正确。

**严重程度：** 次要

**权变措施：** 轮询状态寄存器变化，直到它反映已编程的 SRC\_SEL 和 DIV 值。

### **i2348** *VDDA1V8 静态功率泄漏*

---

**详细信息：** 当 DACVREF 接地时，器件启动后 VDDA1V8 存在静态泄漏。

**权变措施：** 需将 DAC 基准电压与 VDDA1p8V 短接在一起。

### **i2364** *QSPI：在存储器映射模式下不支持访问超过 8MB 的地址*

---

**详细信息：** 从 SoC 互连到 QSPI 控制器的地址行为 23。因此，这将 QSPI 闪存的使用限制为 memmap 模式下每个芯片选择 8MB。

**权变措施：**  
  
无

### **i2508** *安全系统中 RC OSC 的使用*

---

**详细信息：** 内部 10MHz RC 振荡器用作 XTAL 时钟的安全监视器，检测其存在与否，具体如 TRM 的“跛行模式”一节所述。若 XTAL 发生故障，器件会自动切换至 RC\_CLK，以维持 CPU 和外设的运行。由于 RC 振荡器的精度无法与 XTAL 相比，因此它只能检测 XTAL 时钟是否存在（是否翻转），而无法验证 XTAL 时钟的精度或频率精度。

启用跛行模式：

- 跛行模式默认处于禁用状态，并且需由软件显式启用
- XTAL 时钟丢失检测：设置 MSS\_TOPRCM\_LIMP\_MODE\_EN 寄存器中的 LIMP\_MODE\_EN\_XTALCLK\_LOSS\_EN 位。

**权变措施：**  
  
如需监控 XTAL 时钟或 PLL 时钟的精度，需向 MCU 提供外部高精度基准时钟。然后可配置 DCC（双时钟比较器）模块，以使用此外部基准时钟来验证 XTAL 或 PLL 时钟的精度。

### 3 器件公告

#### i2310

#### ***USART: 错误地清除/触发超时中断***

---

详细信息：

在 RHR/MSR/LSR 寄存器读取期间，USART 可能会错误地清除或触发超时中断。

权变措施：

#### **CPU 用例**

- 如果超时中断被错误地清除：
  - 清除是有效的，因为 FIFO 中的挂起数据会重触发超时中断
- 如果错误地设置了超时中断，并且 FIFO 为空，请使用以下软件权变措施来清除中断：
  - 在 TIMEOUTH 和 TIMEOCTL 寄存器中设置超时时计数器的高值
  - 将 EFR2 第 6 位设置为 1 以将超时模式更改为周期性模式
  - 读取 IIR 寄存器以清除此中断
  - 将 EFR2 第 6 位设置回 0 以将超时模式改回原始模式

#### **DMA 用例**

- 如果超时中断被错误地清除：
  - 清除是有效的，因为下一个周期性事件会重触发超时中断
  - 用户必须通过将 EFR2 第 6 位设置为 1 来确保 RX 超时行为处于周期性模式
- 如果超时中断被错误地设置：
  - 这会导致 DMA 被软件驱动程序关闭
  - 有效，因为下一个传入数据将导致软件再次设置 DMA

#### i2311

#### ***USART 虚假 DMA 中断***

---

详细信息：

当在 TLR 寄存器中出现非 2 的乘幂触发级别的情况下，使用 DMA 访问 TX/RX FIFO 时，可能会发生虚假 DMA 中断。

权变措施：

为 TX/RX FIFO 触发级别使用 2 的乘幂值 ( 1、2、4、8、16 和 32 )。

#### i2313

#### ***GPMC : NAND 和 FPGA/FIFO 的低于 32 位读取问题***

---

详细信息：

在 GPMC 接口上进行低于 32 位的读取操作时，会丢失部分数据，从而导致读取到的数据不正确。此情况包括从 NAND 器件或 FPGA/FIFO 接口进行 8 位或 16 位读取。请注意，GPMC 接口不允许 3 字节访问。

权变措施：

GPMC 接口上的读访问必须以 32 位读取方式进行。写入操作不受此项勘误影响。

**i2329****MDIO : MDIO 接口损坏 ( CPSW 和 PRU-ICSS )****详细信息 :**

CPSW 和 PRU-ICSS 外设的所有实例 ( 如果存在 ) 的 MDIO 接口可能在 MDIO 读取时返回损坏的读取数据 ( 例如返回过时数据或以前的数据 ) 或在 MDIO 写入时发送错误的数  
据。MDIO 接口也可能在下次外设复位 ( 通过 LPSC 或在 CPSW 的情况下禁用复位隔离时  
通过全局器件复位 ) 之前不可用。

该问题在系统层面的表现可能包括 (1) 错误的以太网 PHY 链路断开状态 (2) 无法通过  
MDIO 正确配置以太网 PHY (3) PHY 检测不正确 ( 例如地址错误 ) (4) 尝试通过 MDIO 配  
置 PHY 时读取或写入超时。

对于启动模式 ( 如果支持, 仅限 CPSW ), 不存在确保主以太网启动成功的权变措施。如  
果在主启动期间发生该异常, 启动可能发起重试, 重试可能成功, 也可能不成功。如果重  
试不成功, 这将导致最终超时并转换到备用启动模式 ( 如已选定 )。如未选定备用启动模  
式, 此类故障将导致超时并强制通过芯片看门狗进行器件复位, 之后整个启动过程将再次  
重启。

要选择备用启动选项 ( 如果支持 ), 请在启动模式引脚上安装适当的拉电阻器。请参阅各  
特定器件选项的启动文档, 通过以太网进行主启动尝试的典型超时为 60 秒。

**权变措施 :**

在受影响的器件上, 应使用以下权变措施 :

**MDIO 手动模式 : 适用于 PRU-ICSS 和 CPSW。**

可以通过读取和写入 MDIO 外设 MDIO\_MANUAL\_IF\_REG 寄存器中的相应位来模拟  
MDIO 协议, 直接操作 MDIO 时钟和数据引脚。有关手动模式寄存器位及其功能的完整详  
细信息参见 TRM。

在这种情况下, 器件引脚多路复用应配置为允许 IO 由 CPSW 或 PRU-ICSS 外设控制 ( 与  
正常预期运行相同 ), 但必须确保 MDIO\_CONTROL\_REG 中的  
MDIO\_CONTROL\_REG.ENABLE 位为 0 来禁用 MDIO 状态机, 同时将  
MDIO\_POLL\_REG.MANUALMODE 位设置为 1 来启用手动模式。

联系 TI 了解如何实施软件权变措施。

**备注**

如果使用以太网 DLR ( 设备层环网 ) ( 在 CPSW 或 PRU-ICSS 上 ) 或  
EtherCAT 协议 ( 在 PRU-ICSS 上 ), 由于链路状态检查需要轮询间隔, 实施运  
行时权变措施 1 可能对 CPU 或 PRU 负载产生重大影响。因此应考虑系统产生  
的影响。

对于 PRU-ICSS, 使用 MDIO 的 MLINK 功能通过 MIIx\_RXLINK 输入引脚到 PRU-ICSS 自  
动轮询链路状态可以减少软件权变措施的负载, 其中 PRU-ICSS 必须连接外部来自 PHY  
且链路处于活动状态时不会切换的状态输出。根据外部 PHY 器件的指定行为, 该 PHY 状态  
输出可能是 LED\_LINK 或 LED\_SPEED, 也可能是 LED\_LINK 和 LED\_SPEED 的逻辑  
“或”。有关使用 MDIO 的 MLINK 功能的详细信息, 请参阅 TRM 的 MDIO 部分。此功能  
在 CPSW 外设上不可用。

对于在 PRU-ICSS 上实施 EtherCAT, 软件权变措施将在 RTUx/ TX\_PRUx 核心中完成。  
核心必须专门针对权变措施, 即无法用于其他目的。实施方案将支持通过两个用户访问通  
道访问 MDIO。这为 R5f 核心和 PRU 核心提供独立访问通道的选项。这些 API 将与我们在  
RTOS 权变措施实施中的 API 类似。

EtherCAT 将继续通过链路状态的 MDIO MLINK 绕过状态 m/c ( 此路径不受勘误表影响 )  
使用 PHY 快速链路检测。这样可以确保仍然满足电缆冗余相关延迟要求。

i2329 (续)

MDIO : MDIO 接口损坏 (CPSW 和 PRU-ICSS)

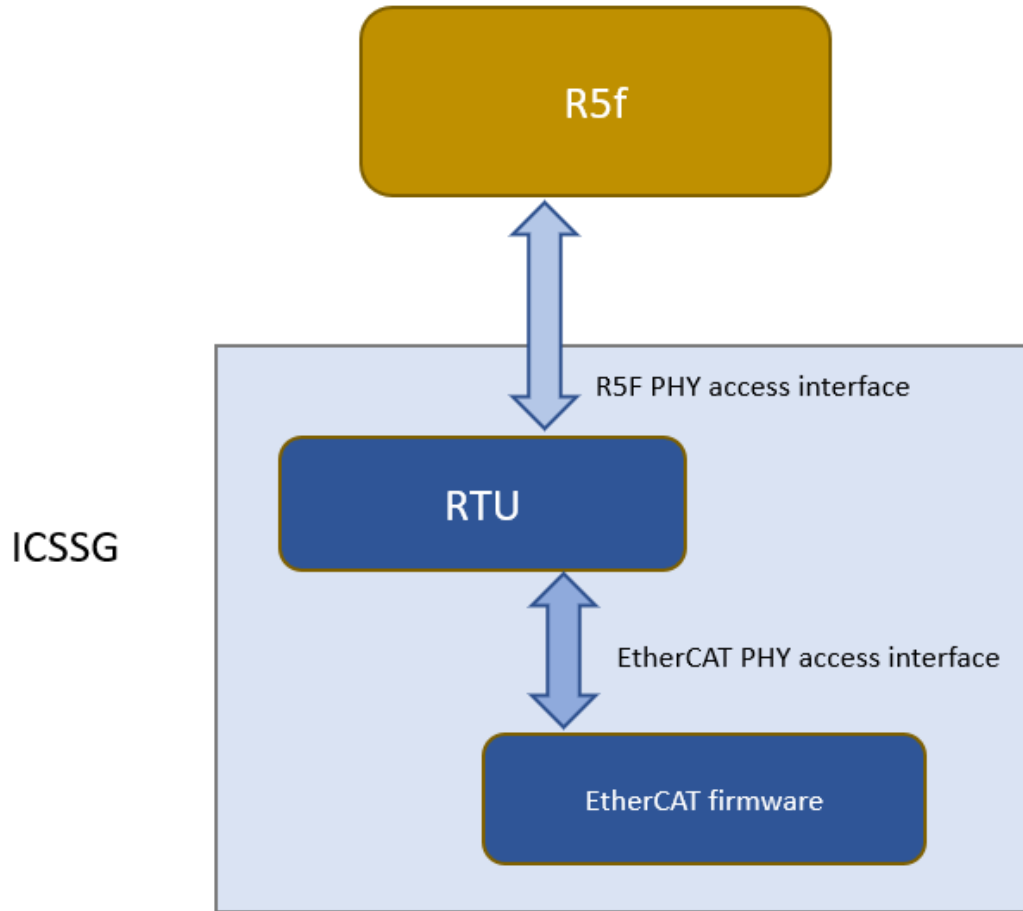


图 3-1. 使用 PRU 核心通过手动模式进行 MDIO 模拟

i2345

CPSW : 如果 CPDMA 提取跨越存储器组的数据包，则会发生以太网数据包损坏

详细信息：

SoC 中的每个存储器组都有一个单独的存储器控制器。即使内存地址是连续的，每个存储体也是具有单独控制器的单独实体。

如果存储器组接收到的存储器请求表示为 32 字节，并且存储器请求的地址在存储器组结束之前 16 字节，则存储器控制器的行为将为：

当存储器控制器在 16 字节之后遇到存储器组末尾时，它将环绕并从存储器组的开头提供 16 个字节。

这会导致数据包损坏。

权变措施：

确保从应用端单个以太网数据包不跨越存储器组。

**i2346** *在奇数通道和偶数通道之间切换时 ADC 结果出现误差***详细信息：**

当 ADC 转换序列涉及对奇数通道和偶数通道进行采样时，每当从奇数通道切换到偶数通道或从偶数通道切换到奇数通道时，转换结果均会出现误差。

仅在偶数通道之间或仅在奇数通道之间进行切换的转换不会出现误差。

**权变措施：**

奇偶数通道切换后的首次采样需采用最小的采集窗口，且该次结果需忽略。

**i2347** *ADC 的 VREF 电流消耗在上电时为随机***详细信息：**

PORZ 后 VREF 电流消耗较高 (1.6mA)，通过 MMR 使能 ADC 后电流降低。

每次 PORZ 周期中，初始电流消耗随机变化。

**权变措施：**

切勿在不复位 DTC 的情况下禁用 ADC。

**i2349** *断电时 ADC VrefHi 负载增加***详细信息：**

若 ADC 在转换完成后被禁用，基准电压上的负载将增加 2mA。

**权变措施：**

切勿在不复位 DTC 的情况下禁用 ADC。

**i2350** *McSPI：在“ABS SYNC”模式下使用 EDMA 的 McSPI 数据传输在 32 位传输后停止***详细信息：**

若将 EDMA 配置为向 McSPI Tx FIFO ( 32 字节 ) 传输超过 32 位数据，则其仅在将前 32 位数据传输到 FIFO 后停止工作。

此问题仅在 EDMA 的“ABS SYNC”模式下观察到，即 EDMA 配置的传输大小超过 32 位。

此问题发生时，EDMA 既不会传输数据并完成传输，也不会引发任何错误，因为 McSPI 未针对 EDMA 的事务产生 vbusp\_sdones 信号。

SPI RX 模式不受此问题影响。

**权变措施：**

选项 1：对于 McSPI TX 操作，请使用 EDMA 的 ASYNC 模式

选项 2：若 McSPI TX 操作需使用 ABS SYNC 模式，则设置 acnt=4、bcnt=1 且 ccnt=1

**i2352** *CONTROLSS-SDFM：动态更改阈值设置 (LLT、HLT)、滤波器类型或 COSR 设置将触发虚假比较器事件***详细信息：**

当 SDFM 比较器设置 ( 例如滤波器类型、下限/上限阈值或比较器 OSR (COSR) 设置 ) 在运行期间动态更改时，将触发虚假比较器事件。如果配置得当，虚假比较器事件将触发相应的 CPU 中断、CLA 任务、ePWM X-BAR 事件和 GPIO 输出 X-BAR 事件。

**权变措施：**

当需要动态更改比较器设置时，请执行以下程序，以确保虚假比较器事件不会产生 CPU 中断、CLA 事件或 X-BAR 事件 ( ePWM X-BAR/GPIO 输出 X-BAR 事件 )：

**i2352 (续)**

**CONTROLSS-SDFM : 动态更改阈值设置 (LLT、HLT)、滤波器类型或 COSR 设置将触发虚假比较器事件**

---

1. 禁用比较器滤波器。
2. 至少延迟比较器滤波器的延时 + 3 个 SD-Cx 时钟周期。
3. 更改比较器滤波器设置, 例如滤波器类型、COSR 或下限/上限阈值。
4. 至少延迟比较器滤波器的延时 + 5 个 SD-Cx 时钟周期。
5. 启用比较器滤波器。

**i2353**

**CONTROLSS-SDFM : 动态更改数据滤波器设置 (例如滤波器类型或 DOSR) 将触发虚假数据响应事件**

---

**详细信息:**

当 SDFM 数据设置 (例如滤波器类型或 DOSR 设置) 在运行期间动态更改时, 将触发虚假数据滤波器就绪事件。如果配置得当, 虚假数据就绪事件将触发相应的 CPU 中断、CLA 任务和 DMA 触发器。

**权变措施:**

当需要动态更改 SDFM 数据滤波器设置时, 请执行以下程序, 以确保不会产生虚假数据滤波器就绪事件:

1. 禁用数据滤波器。
2. 至少延迟数据滤波器的延时 + 3 个 SD-Cx 时钟周期。
3. 更改数据滤波器设置, 如滤波器类型和 DOSR。
4. 至少延迟数据滤波器的延时 + 5 个 SD-Cx 时钟周期。
5. 启用数据滤波器。

**i2354**

**CONTROLSS-SDFM : 在三个 SD 调制器时钟周期内对 SDCPARMx 寄存器位字段 CEVT1SEL、CEVT2SEL 和 HZEN 进行两次背对背写入可能会损坏 SDFM 状态机, 从而导致虚假比较器事件**

---

**详细信息:**

在三个 SD 调制器时钟周期内对 SDCPARMx 寄存器位字段 CEVT1SEL、CEVT2SEL 和 HZEN 进行背对背写入可能会损坏 SDFM 状态机, 从而导致虚假比较器事件, 如果配置得当, 这可能触发 CPU 中断、CLA 任务、ePWM XBAR 事件和 GPIO 输出 X-BAR 事件。

**权变措施:**

避免在三个 SD 调制器时钟周期内背对背写入或在一次寄存器写操作中配置 SDCPARMx 寄存器位字段。

**i2355**

**CONTROLSS-ADC : DMA 读取过时结果**

---

**详细信息:**

ADCINT 标志可在 ADCRESULT 值锁存前置位 (参见 AM263x 技术参考手册中 ADC 时序表的 tLAT 和 tINT(LATE) 列)。

DMA 可在 ADCINT 触发置位后仅三个周期就读取 ADCRESULT 值。因此, 若满足以下所有条件, 当用户期望获得最新结果时, DMA 可能读取到之前的 ADCRESULT 值:

- ADC 处于延迟中断模式。
- ADC 工作于 tINT (LATE) 比 tLAT 早三个或更多周期发生的模式下 (12 位模式下 ADCCTL2 [PRESCALE] > 2)。
- DMA 由 ADCINT 信号触发。

**i2355 (续)**
**CONTROLSS-ADC : DMA 读取过时结果**

- DMA 在未先读取任何其他值的情况下，立即读取与该 ADCINT 信号关联的 ADCRESULT 值。
- DMA 在收到 ADCINT 触发时处于空闲状态。

仅上述 DMA 读取可能导致读取到陈旧数据；以下非 DMA 方法将始终读取预期数据：

- ADCINT 标志触发 CLA 任务。
- ADCINT 标志触发 CPU ISR。
- CPU 轮询 ADCINT 标志。

**权变措施：**

从 ADCINT 标志触发两个 DMA 通道。第一个通道作为虚拟事务。这将产生足够的延迟，使得第二个通道总能读取到最新的 ADC 结果。

**i2356**
**CONTROLSS-ADC : 如果未设置 INTxCONT (继续中断模式)，中断可能会停止**
**详细信息：**

在 ADCINTSELxNx[INTxCONT]= 0 时，设置 ADCINTFLG 后，中断将停止，并且不会发生其他 ADC 中断。若在 ADCINTFLGCLR 寄存器进行软件写入的同时发生 ADC 中断，则 ADCINTFLG 将意外保持为设置状态，阻止将来发生 ADC 中断。

**权变措施：**

1. 使用“继续中断”模式，则 ADCINTFLG 无法阻止其他 ADC 中断：

```
ADCINTSEL1N2[INT1CONT] = 1;
ADCINTSEL1N2[INT2CONT] = 1;
ADCINTSEL3N4[INT3CONT] = 1;
ADCINTSEL3N4[INT4CONT] = 1;
```

2. 为了避免发生这种情况，请确保下一次发生 ADC 中断之前，始终有足够的时间为 ADC ISR 提供服务并清除 ADCINTFLG。
3. 清除 ADCINTFLG 时，请检查 ISR 中是否存在溢出情况。在写入到 ADCINTFLGCLR 后立即检查 ADCINTOVF；如果已设置，则再次写入 ADCINTFLGCLR 以确保 ADCINTFLG 已被清除。若再设置 ADCINTOVF 寄存器，则表示已丢失 ADC 转换中断。

```
AdcaRegs.ADCINTFLGCLR.bit.ADCINT1 = 1; //clear INT1 flag
if(1 == AdcaRegs.ADCINTOVF.bit.ADCINT1) //ADCINT overflow
{
    AdcaRegs.ADCINTFLGCLR.bit.ADCINT1 = 1; //clear INT1 again
    // If the ADCINTOVF condition will be ignored by the application
    // then clear the flag here by writing 1 to ADCINTOVFCLR.
    // If there is a ADCINTOVF handling routine, then either insert
    // that code and clear the ADCINTOVF flag here or do not clear
    // the ADCINTOVF here so the external routine will detect the
    // condition.
    // AdcaRegs.ADCINTOVFCLR.bit.ADCINT1 = 1; // clear OVF
```

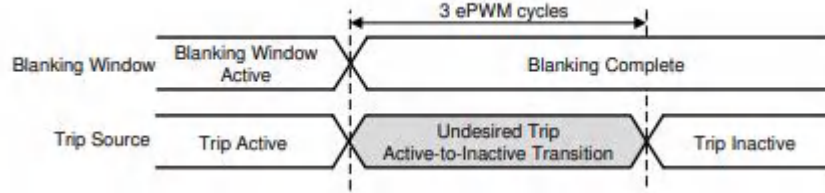
**i2357**
**CONTROLSS-ePWM : 如果跳闸在消隐窗口结束时保持活动状态，则可能会发生 ePWM 干扰**
**详细信息：**

消隐窗口通常用于屏蔽转换期间将导致系统误跳闸的任何 PWM 跳闸事件。如果在消隐窗口周期结束后，ePWM 跳闸事件在少于三个 ePWM 时钟内保持活动状态，则 ePWM 输出上可能会存在不良干扰脉冲。

以下图片展示了可能会导致不良 ePWM 输出的时间段。

i2357 (续)

**CONTROLSS-ePWM : 如果跳闸在消隐窗口结束时保持活动状态, 则可能会发生 ePWM 干扰**



以下图片展示了如果跳闸事件在消隐窗口关闭之前的 1 个周期或关闭之后的 3 个周期内结束, 可能出现的两个 ePWM 输出。



权变措施 :

避免配置消隐窗口, 使跳闸输入处于此范围内 (消隐窗口关闭前 1 个周期和后 3 个周期)。

i2358

**CONTROLSS-ePWM : 在消隐开始后的前 3 个周期内, 消隐窗口不会过滤跳闸事件**

详细信息 :

在启动消隐窗口后的前 3 个周期内, 消隐窗口不会消隐跳闸事件。DCEVTFILT 可能继续反映 DCxEVty 信号的变化。如果启用 DCEVTFILT, 则可能会影响配置的后续子系统 (例如跳闸区域子模块、TZ 中断、ADC SOC 或 PWM 输出)。

权变措施 :

在需要消隐之前的 3 个周期启动消隐窗口。如果在周期边界需要消隐窗口, 则在下一周期开始前的 3 个周期启动消隐窗口。这是因为消隐窗口在整个期间边界中持续存在。

i2359

**CONTROLSS-CMPSS : 当 DACSOURCE 设为 0 或重新配置为 1 时, 预分频器计数器的行为与规范不同**

详细信息 :

在预分频器运行时, 如果我们使 DACSOURCE = 0, 预分频计数器将不会复位, 如果启用条件为低电平, 该值仍然保持不变; 当 DACSOURCE 再次配置为 1 时, 该计数器从之前被保留的值开始计数。只有在预分频计数器运行期间配置 DACSOURCE 时, 才会出现该错误。

权变措施 :

在 DACSOURCE 配置 (非动态配置) 之间发出软复位。

i2374

**如果 R5SS\_CORE\_CLK 的时钟频率与 R5FSS\_CLK\_SELECTED 频率不同, 则 PBIST 失败**

详细信息

R5SS 存储器接收 R5SS CPU 时钟 "R5SS\_CORE\_CLK", 该时钟使用可编程分频器从 R5SS\_CLOCK\_SELECTED 根时钟派生而来。使用 PBIST 控制器测试 R5SS 存储器时, PBIST 控制器会接收 R5SS\_CLOCK\_SELECTED 根时钟。如果为两个时钟选择不同的频率, 则 PBIST 操作会失败。

**i2374 (续)**

**如果 R5SS\_CORE\_CLK 的时钟频率与 R5FSS\_CLK\_SELECTED 频率不同，则 PBIST 失败**

**权变措施**

为了使 PBIST 与 R5SS 存储器配合使用，两个时钟的频率需要相同。如果应用使用要求 R5SS\_CORE\_CLK 为 R5SS\_CLOCK\_SELECTED 的分频频率，那么在 R5SS 存储器的 PBIST 操作期间，应用应确保 R5SS\_CORE\_CLK 配置为与 R5SS\_CLOCK\_SELECTED 相同的频率。

**i2375**

**如果比较器事件仍处于活动状态并且正在选择数字滤波器路径 (使用 SDCOMPxCTL.CEVTxDIGFILTSEL)，则不会再次设置 SDFM 模块事件标志 (SDIFLG.FLTx\_FLG\_CEVTx)**

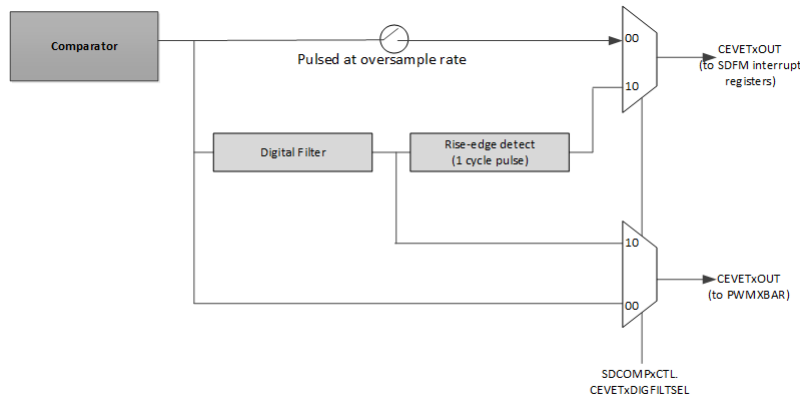
**详细信息**

SDFM 模块支持在 SDFM 比较器输出端配置数字滤波器，该滤波器可由应用程序选择用于滤除毛刺干扰。应用程序可选择比较器的滤波输出或原始输出，以到达事件标志寄存器 (SDIFLG.FLTx\_FLG\_CEVTx) 和 SDFM 模块的 CEVETxOUT 事件输出，如图所示。从数字滤波器到事件标志寄存器的路径上有一个上升沿检测逻辑，如图所示。

选用数字滤波器路径时，事件标志寄存器仅在数字滤波器输出的上升沿置位一次。若事件标志寄存器清零，即使比较器输出保持高电平，该寄存器也不会再次置位。

此问题不会出现在发往 XBAR 的 CEVETxOUT 事件上。

若选择原始输出路径 (即 CEVTxDIGFILTSEL = 0)，此问题也不存在。



**权变措施**

若应用中使用 SDFM 数字滤波器，可考虑以下解决方法：

- **选项 1**
  - 可观察 XBAR 状态，而不是事件标志寄存器。
- **选项 2**
  1. 选择数字滤波器后，等待中断/跳闸。
  2. 发生中断时，读取事件标志并采取适当的应用操作，以消除导致比较器跳闸的诱因。
  3. 清除事件标志寄存器之前，先配置非滤波路径。
  4. 清除事件标志。
  5. 读取事件标志，若其至少在一个过采样周期内保持清零状态，则重新配置数字滤波器路径。

**i2375 (续)**

**如果比较器事件仍处于活动状态并且正在选择数字滤波器路径 (使用 `SDCOMPxCTL.CEVTxDIGFILTSEL` ) , 则不会再次设置 `SDFM` 模块事件标志 (`SDIFLG.FLTx_FLG_CEVTx`)**

**备注**

执行第二至第四步操作期间, PWM 跳闸逻辑也将基于未滤波的 `SDFM` 比较器输出工作。

**i2386**

**`CRC` : 不支持 `CRC 8` 位数据宽度以及 `CRC8-SAE-J1850` 和 `CRC8-H2F` 在 `CAN` 模块中的潜在用途**

**详细信息 :**

`CRC` 类型 `CRC8-SAE-J1850` 和 `CRC8-H2F` 不支持 `8` 位数据宽度。支持的最小数据宽度为 `16` 位。

**权变措施 :**

无权变措施。建议避免使用上述不受支持的多项式。

**i2392**

**`mem-init` 捕获寄存器中的竞争条件导致事件丢失**

**详细信息 :**

捕获寄存器中的潜在竞态条件导致事件丢失, 而写入寄存器将清除同一寄存器中的其他事件。以下寄存器受此问题的影响:

`MSS_CTRL` : `*MEMINIT_DONE` 寄存器

**权变措施 :**

可以使用以下任意权变措施:

按顺序触发 `mem-init` 并清除状态, 然后触发新的 `mem-init`。两个状态均在同一个寄存器中时需要执行此操作。

( 或者 )

如果必须并行触发, 必须轮询所有触发为 `1'b1` 的状态位, 然后转到并清除 `DONE` 状态寄存器

( 或者 )

启动 `mem-init` 后检查 `MEM_INIT_STATUS` 寄存器, 定期检查状态等待状态变为低电平, 最后在状态变为低电平时清除 `DONE` 状态寄存器

**i2393**

**`BUS_SAFETY_ERR` 寄存器中未记录检测到的故障的粒度错误状态**

**详细信息 :**

未在 `MSS_CTRL:*_BUS_SAFETY_ERR` 寄存器的 `COMP_CHECK` 和 `COMP_ERR` 字段中正确记录检测到的故障的粒度错误状态。

错误信号 `err_comp` 和 `err_comp_signals` 用于检测诊断电路上的任何故障。这两个信号的 `AND` 输出用于报告故障。但是, 它们在时钟的不同边沿进行采样, 导致生成的错误信号丢失, 因此不会记录在 `MSS_CTRL` MMR 中。

有两种可能的情况:

情况 1 : 对数寄存器具有非零值

此处正确捕获了细化日志, 并可对给定的故障执行相应操作。

i2393 (续)

**BUS\_SAFETY\_ERR 寄存器中未记录检测到的故障的粒度错误状态**

情况 2：日志寄存器全部为零

此处未正确捕获粒度日志，可能受影响的实体是 R5F 和 L2 存储器。

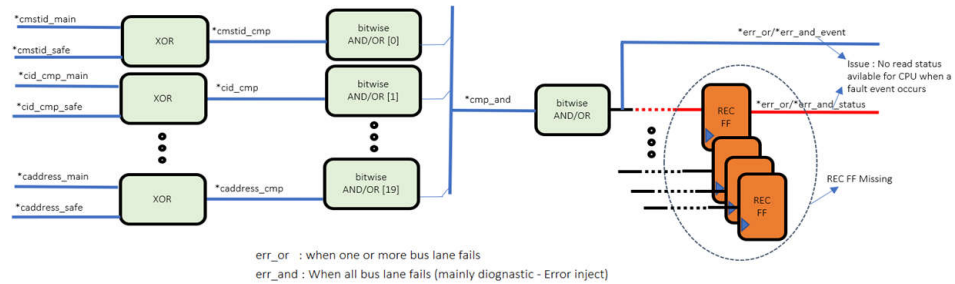


图 3-2.

权变措施：

无权变措施。

由于未记录粒度错误状态，因此总线安全故障仅作为聚合的错误事件进行检测，并且无法正确捕获诊断信息的粒度。

情况 1：在正确捕获日志的情况下，可以使用日志结果来采取适当的操作。

情况 2：如果未正确捕获日志，则 a) 诊断不需要执行任何操作；b) 如果应用程序中实际发生故障，则应使用 WarmResetn。

i2394

**中断和错误聚合器捕获寄存器中的竞争条件导致事件丢失**

详细信息：

捕获寄存器中的潜在竞态条件导致事件丢失，而写入寄存器将清除同一寄存器中的其他事件。以下寄存器受此问题的影响：

MSS\_CTRL : \*INTAGG\_STATUS\_REG、\*TPCC\_ERR/INTAGG\_STATUS\_RAW

权变措施：

在 ISR 中执行以下步骤：

- 1) 在退出 ISR 之前读取 \*\_ERRAGG\_RAW 并通过 \*\_ERRAGG\_MASK 的与操作检查位有效性。
- 2) 如果设置任何位，此时表示在清除 \*\_ERRAGG\_STATUS 时中断/错误丢失。
- 3) 在 ISR 中处理相应的位，然后退出 ISR。应在 STATUS 和 “RAW&MASK” 为零后退出 ISR

i2395

**DCC 模块频率比较可报告错误结果**

详细信息：

双时钟比较器模块用于监测时钟频率并与已知时钟基准对比，它可能比预期更早停止，从而指示被测时钟频率偏低。这是由于时钟域交叉问题导致错误检测逻辑的预设被触发。

权变措施：

**解决方法 (1)：**在可能的情况下，应用程序代码可使用备用 EDCC 模块（存在于 MSS 中）来比较时钟

**i2395 (续)**

***DCC 模块频率比较可报告错误结果***

---

**解决方法 (2) :** 可对同一时钟对进行多次测量，并忽略报告的异常频率。

**i2401**

***CPSW : 主机时间戳导致 CPSW 端口锁定***

---

**详细信息 :**

CPSW 提供两种机制，将数据包进站时间戳信息传递给主机。

第一种机制是通过 CPTS 事件 FIFO 实现的，该机制在由特定事件触发时记录时间戳。这类事件包括接收到包含指定 EtherType 字段的以太网数据包。这种机制最常用于捕获 PTP 数据包的进站时间戳。通过这种机制，主机必须将时间戳（来自 CPTS FIFO）和 DMA 提供的数据包有效载荷分开读取。这种模式受支持并且不受此勘误表影响。

第二种机制是为所有数据包启用接收时间戳，而不仅仅是 PTP 数据包。采用这种机制时，时间戳会通过 DMA 与数据包有效载荷一起传送。第二种机制是本勘误表的主题。

当启用 CPTS 主机时间戳时，每个传送到内部 CPSW 端口 FIFO 的数据包都需要来自 CPTS 的时间戳。当数据包前导码由于 EMI 或任何其他损坏机制而损坏时，可能不会向 CPTS 发送时间戳请求。在这种情况下，CPTS 将不会生成时间戳，这会导致 CPSW 端口 FIFO 中出现锁定状况。当通过清除 CPTS\_CONTROL 寄存器中的 `tstamp_en` 位来禁用 CPTS 主机时间戳时，将阻止发生锁定状况。

**权变措施 :**

必须禁用以太网到主机的时间戳。

可以使用 CPTS 事件 FIFO 时间戳来代替 CPTS 主机时间戳。

**i2402**

***CPSW : 以太网到主机校验和卸载不起作用***

---

**详细信息 :**

启用以太网转主机校验和功能存在问题，会使 CPSW 进入不可恢复的错误状态。主机转以太网校验和功能不受该问题影响。

**权变措施 :**

无。不得启用 `P0_TX_CHKSUM_EN`。

**i2403**

***M4 ROM : 在 HSSE 器件上不支持 SBL 冗余启动映像特性***

---

**详细信息 :**

在 HSSE 器件上不支持 SBL 冗余启动映像特性

如果主映像以下位置发生任何损坏，SBL 引导将无法从冗余闪存区域引导

- 证书中间的图像损坏
- 证书结束时图像损坏
- SBL 二进制文件开始时图像损坏
- SBL 二进制文件中间的图像损坏
- SBL 二进制文件结束时图像损坏

**权变措施 :**

无。

**i2404*****MBOX : 邮箱寄存器中的竞争条件导致事件丢失*****详细信息 :**

捕获寄存器中的潜在竞态条件导致事件丢失，而写入寄存器将清除同一寄存器中的其他事件。以下寄存器受此问题的影响：

MSS\_CTRL : \*\_MBOX\_READ\_REQ

MSS\_CTRL : \*\_MBOX\_READ\_DONE

**权变措施 :**

在设置触发 (WRITE DONE /READ ACK) 事件之前，读取其他处理器的状态 (READ DONE / READ\_DONE\_REQ) 来检查中断是否处于执行状态。

( 或者 )

如果未在给定时间内接收到状态 (READ DONE / READ\_DONE\_REQ)，再次触发 (WRITE DONE /READ ACK) 事件。

**i2405*****CONTROLSS : 竞争条件 OUTPUT\_XBAR 和 PWM\_XBAR 导致事件丢失*****详细信息 :**

捕获寄存器中的潜在竞态条件导致事件丢失，而写入寄存器将清除同一寄存器中的其他事件。以下寄存器受此问题的影响：

C2K\_PWMXBAR:PWMXBAR\_STATUS

C2K\_OUTPUTXBAR:OUTPUTXBAR\_STATUS

**权变措施 :**

WA -1 ( 适用于大于 50ns 的事件宽度 ) :

默认情况下，将在“状态”寄存器中捕获电平事件 ( 宽度 >50ns )，同时执行“清除标志”，如果硬件中的任何新事件同时置为有效，则新事件将在“标志”寄存器中丢失，但是，状态寄存器会捕获标志寄存器中丢失的此类事件。完成“清除标志”后，读取“状态”寄存器可以根据读取的“状态”来捕获/处理任何错过的事件。

WA-1 : ISR 序列 :

读取标志事件[x]

读取状态，所有事件

清除标志，事件[x]

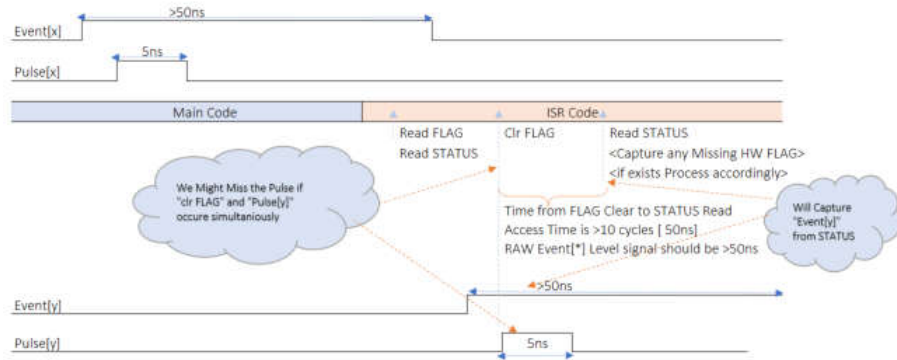
读取状态，所有事件

捕获任何缺失的硬件事件标志

如果存在，则进行相应处理

i2405 (续)

**CONTROLSS : 竞争条件 OUTPUT\_XBAR 和 PWM\_XBAR 导致事件丢失**



WA -2 ( 适用于任何事件宽度 ) :

在 ISR 中使用相同的事件启用 OUTPUTXBAR ，然后 “清除 PWMXBAR 标志” 。

同一窗口期间缺失的任何硬件事件都将捕获在 OUTPUTXBAR 标志中。读取 OUTPUTXBAR 标志并进行相应处理

在 ISR 中，先 “清除 OUTPUTXBAR 标志” ，然后禁用 OUTPUTXBAR。

WA-2 : ISR 序列 :

读取标志事件[x]

读取状态，所有事件

启用 OutPutxBAR

- 映射相同的事件

清除标志，事件[x] PWMXBAR

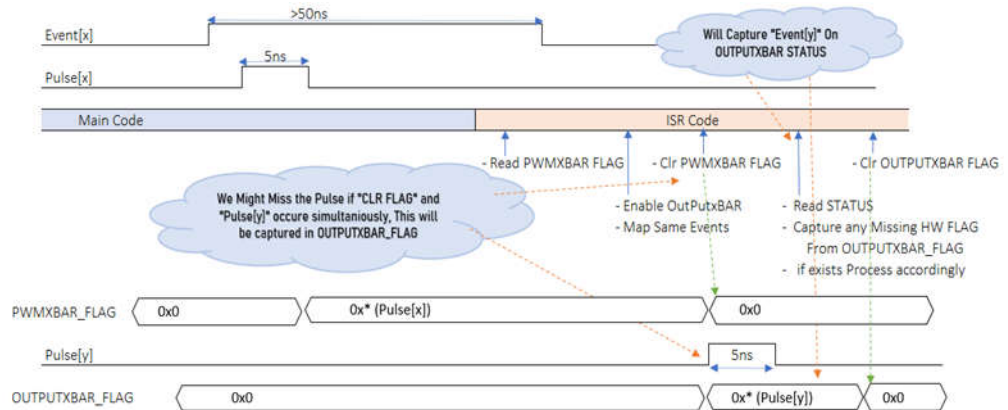
读取状态

- 捕获任何丢失的硬件事件标志

来自 OUTPUTXBAR\_FLAG

如果存在，则进行相应处理

- 清除标志，事件[y] OutputXBAR



## i2427

**RAM SEC 可能会导致虚假的 RAM 写入，从而导致 L2 和 MBOX 存储器损坏**

## 详细信息：

如果存储器在 RAM 读取数据期间由于读取或部分写入事务遇到一个 single-bit 错误，则 RAM 将进入一种状态，如果下一个“存储器读取”是由后续的部分写入事务导致的，则可能会导致稍后向 RAM 进行虚假写入。相反，如果“存储器读取”是由实际的存储器读取事务导致的，则将清除挂起的不良内部状态，并且以后不可能发生虚假写入。虚假写入是指在触发虚假写入的部分写入事务之前写入的最后一个存储器地址。此问题仅适用于 MBOX & L2。

为更加清晰起见，图 3-3 列出了问题适用（例如 1、2 和 3）和不适用（例如 4、5 和 6）的可能情形。Transaction# 仅用于说明，并不一定表示每次操作发生的确切周期。[SEC - 单位错误校正、DED - 双位错误检测]

Ex #	Transaction 1	Transaction 2+N N=0,1,2,3..	Transaction 2+N+1	Transaction 2+N+2
1	Read or Partial Write Addr A (SEC) ← read with SEC	Full Write Addr X ← last write prior to partial write Note: N=0	Partial Write ← Triggers spurious write	Spurious write to Addr X with Transaction 1 corrected read data of Addr A
2	Read or Partial Write Addr A (SEC) ← read with SEC	Full Write Addr B Full Write Addr C Full Write Addr D ← last write prior to partial write Note: N=2	Partial Write ← Triggers spurious write	Spurious write to Addr D with Transaction 1 corrected read data of Addr A
3	Read or Partial Write Addr A (SEC)	Partial Write Addr B Note: N=0	Spurious write to Addr A with Transaction 1 corrected read data of Addr A (Addr A is overwritten with the RAM content prior to the Transaction 1 Partial write)	
4	Read Addr A (SEC)	Partial Write Addr B Note: N=0	No Spurious write to Addr A with Transaction 1 corrected read data of address A (no data corruption)	
5	Read or Partial Write Addr A (SEC)	Read ← Clears bad internal state Note: N=0	No spurious writes with all command combinations in subsequent cycles	
6	Read or Partial Write Addr A (SEC)	Full Write Addr B Note: N=0	Read ← Clears bad internal state	No spurious writes will all command combinations in subsequent cycles

图 3-3.

## 权变措施：

以下选项之一可用作权变措施。

**选项 1：**

禁用 ECC，仅适用于非安全应用。

**选项 2：**

不允许对存储器进行部分写入（仅执行全行写入）

对于 L2，如果 L2 空间可缓存，则内核将仅执行完整行写入，因此此问题不适用。

**选项 3：**

应用程序可以像 DED 一样处理所有 SEC 错误（即使在 single-bit 错误的情况下也仅检测纠正错误），因为如果应用无法在读取或部分写入事务发生 single-bit 错误后立即控制事务，则可能会损坏 RAM 数据。

**备注**

之前关于使用 ECC CTRL - SEC 计数器作为正常 SEC 问题与虚假写入的指示器的陈述无效。虚假写入后，ECC CTRL SEC 计数器仍可为 1。

## i2428

### *DTHE 中的 AES 在 GCM 加密结束时为 data\_in 生成额外的 dma 请求*

#### 详细信息：

AES 引擎在加密的 GCM 密码模式结束时为数据输入生成一个额外的 DMA 请求。此问题仅适用于使用 AES-GCM 模式的加密，而不适用于 AES-GCM 解密或任何其他分组密码模式（例如 CBC）。

在几个周期之后，额外的 DMA 请求会自行离开（置为无效），而没有向其写入任何数据。

根据系统中 DMA 设置为 AES-GCM 模式的方式，数据包传输结束时的额外 DMA 请求可能会导致下一个数据包发生意外数据传输。

#### 权变措施：

无

## i2433

### *ICSS：读取 LSW 时，读取 64 位 IEP 计时器没有锁定 MSW 逻辑*

#### 详细信息：

当低 32 位数据为 0xFFFFFFFFC 或更高（在 250MHz 上）时、IEPx 64 位时间戳可能不正确。在这种情况下会更新高 32 位值，但较低的值是旧数字。当 IEP 计数器 (IEP\_COUNT\_REG1: IEP\_COUNT\_REG0) 从 ICSS PRU 内核连续读回时，会出现该问题。

#### 示例 1：

第 1 个读数：0x000000D0（高字节）：0xFFFFFFFFC（低字节）

第 2 个读数：0x000000D0（高字节）：0x00000028（低字节）

#### 示例 2：

第 1 个读数：0x000000D7（高字节）：0xFFFFFFFFC（低字节）

第 2 个读数：0x000000D7（高字节）：0x0000002C（低字节）

#### 示例 3：

第 1 个读数：0x000000D6（高字节）：0xFFFFFFFFF0（低字节）

第 2 个读数：0x000000D7（高字节）：0xFFFFFFFFC（低字节）

如上所示，这会导致计时器增量行为变得非单调，或者计时器差异异常大，如示例 3 中所示。这是由从 IEPx 计数器加载 64 位值时出现的 1 周期竞态条件导致的。

#### 权变措施：

注意：这些权变措施存在于 SDK9.2 和更高版本中

使用 C 语言针对 PRU 的权变措施：

```
uint64_t timestamp = (uint64_t) (0x2E0010);
```

/\*权变措施从此处开始\*/

```
if ((timestamp & 0xFFFFFFFF) >= 0xFFFFFFFF)
{
    timestamp = *(uint64_t*) (0x2E0010); }

```

/\*权变措施在此处结束\*/

**i2433 (续)****ICSS : 读取 LSW 时, 读取 64 位 IEP 计时器没有锁定 MSW 逻辑**

使用汇编语言针对 PRU 的权变措施：

```
ldi32 r4, 0xFFFFFFFF ; 0-4 for 250MHz clock
;load 64-bit timestamp to r2:r3
lbc0 &r2, c26, 0x10, 8
qbg0 skip_iep_read_errata. r2, r4
;re-read IEP if IEP_COUNTER_LOW >= 0xFFFF_FFFC
lbc0 &r2, c26, 0x10, 8
skip_iep_read_errata:
```

使用 C 语言针对 R5F、A53 的权变措施：

```
uint64_t getIepTimeStamp64 (void)
{
    uint64_t u64Timestamp1 = (volatile uint64_t)(0x300AE010);
    uint64_t u64Timestamp2 = (volatile uint64_t)(0x300AE010);
    if (u64Timestamp2 > u64Timestamp1)
    {
#ifdef __DEBUG
        if (((u64Timestamp2 >> 32)-(u64Timestamp1 >> 32)) == 1)
        {
            /* HW errata fixed due to picking u64Timestamp1*/
            if ((u64Timestamp2 & 0xFFFFFFFF) >= (u64Timestamp1 & 0xFFFFFFFF))
            {
                DebugP_log ("Errata fixed (1): %llx : %llx\r\n",
                    u64Timestamp1, u64Timestamp2);
            }
        }
#endif
        return u64Timestamp1;
    }
    else
    {
#ifdef __DEBUG
        if ((u64Timestamp2 & 0xFFFFFFFF) < (u64Timestamp1 & 0xFFFFFFFF))
        {
            /* Adjust the IEP MSW in the case running into HW errata
            */
            DebugP_log ("Errata fixed (2): %llx : %llx\r\n", u64Timestamp1,
                u64Timestamp2);
        }
#endif
        /* HW errata fixed due to picking u64Timestamp2*/
        return u64Timestamp2;
    }
}
```

**i2438****CPSW : 使用 VLAN 添加/移除生成主机到以太网校验和**

详细信息：

当在硬件上启用 CPSW 主机到以太网校验和生成并且在以太网出口处添加或删除 VLAN 标记时，从主机到以太网的数据包损坏，并作为具有良好 CRC 的垃圾发送，这是不可接受的。

权变措施：

对于生成校验和的数据包，在以太网出口处不得添加或删除 VLAN 标记。

**i2439****CPSW : 主机到以太网时间戳精度问题**

详细信息：

**i2439 (续)**

***CPSW : 主机到以太网时间戳精度问题***

---

当数据包从主机发送到以太网时，在以太网出口时生成一个时间戳，低 8 位中数据包长度为 0xD5 时会导致时间戳错误。

PTP 消息的使用时间戳不应受到影响，这是因为 PTP 消息通常比 0xD5 数据包长度短很多。

**权变措施：**

仅对于主机 Tx 上的 PTP 消息，才应启用以太网时间戳。

**i2488**

***时钟：用于精确 50-50 占空比时钟的 PLL 配置***

---

**详细信息：**

PLL 中的 VCO 可生成具有不同占空比的输出波形，无法满足系统和外设所需的精确 50-50 占空比要求。

**严重程度：**

次要

**权变措施：**

将 PLL 配置为以目标频率的两倍 (2x) 运行。配置时钟分频器 (HSDIVIDER) 寄存器以对 PLL 输出进行 2 分频。

**i2499**

***突发读取期间发生单错误检测时，返回给主机的数据不正确***

---

**详细信息：**

当正在进行的多拍突发读取操作期间检测到单比特错误时，RAM 控制器会遇到时序问题。虽然 RAM 控制器中的 ECC 模块正确地纠正了存储器中的错误，但 ECC 模块与 RAM 控制器总线协议逻辑之间的握手时序问题导致数据错位，从而使得在后续的读取节拍中，返回给请求主机 (R5F、DMA、PRU-ICSS、CPDMA、HSM-M4) 的数据不正确。

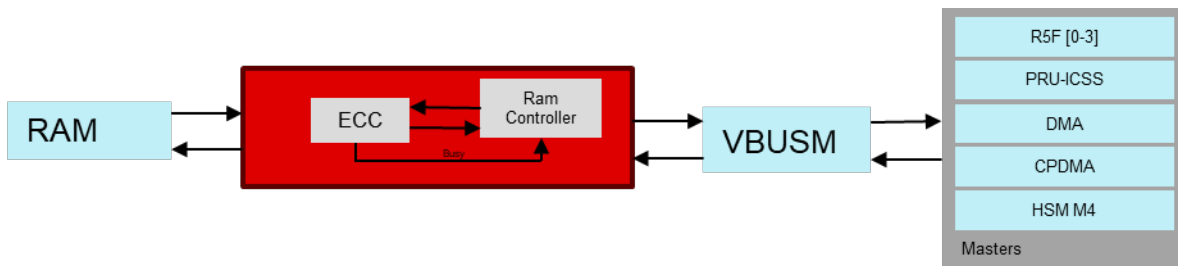
- 在突发读取期间检测到单比特错误时，ECC 模块会在其中插入一个 BUSY 周期。
- 在此 BUSY 周期中，会暂停对存储器的下一次读取，并执行一次修正写回以修复 RAM 中的错误。
- 总线协议逻辑未能正确处理此过程中产生的 BUSY 信号。
- 恢复突发读取后，数据流水线出现错位。

因此：

- 在错误纠正之后的读取操作中，返回给主机的数据不正确。
- 可能导致程序执行失败和异常中止。

此问题仅影响 L2 和 MBOX RAM 的读操作。

物理 RAM 中的内容保持正确纠正状态 (无存储器损坏)，且写入操作功能正常。以下为简要方框图。

**i2499 (续)****突发读取期间发生单错误检测时，返回给主机的数据不正确****权变措施：**

根据系统要求和安全需求，可考虑以下解决方案：

**选项 1：ECC 禁用**

仅适用于非安全应用：彻底关闭错误纠正码 (ECC) 功能。

**选项 2：保守式错误处理**

将系统配置为将所有单比特错误纠正 (SEC) 事件视为双比特错误检测 (DED) 事件。

这意味着即使对于可纠正的单比特错误，也只实现仅检测行为而不尝试纠正。

**选项 3：预防性存储器擦洗**

若上述方案不可行，则定期执行存储器擦洗以主动管理潜在错误：

在应用程序访问之前，系统地读取存储器位置以检测并纠正单比特错误。

实现方法：

- 通过 DMA (直接存储器访问) 进行 SRAM 擦洗  
有关 DMA 擦洗器，请参阅 SDK 参考示例
- 通过 ICSS (工业通信子系统) 进行 SRAM 擦洗  
有关 PRU 擦洗器，请参阅 SDK 参考示例

重要限制：虽然擦洗操作可降低正常运行期间遇到错误的概率，但无法完全消除发生错误的可能性。

**i2500****使用 0xF0 作为唤醒密钥时 LIN 模块无法唤醒****详细信息：**

使用唤醒密钥 0xF0 来唤醒 LIN 模块会导致 LIN 状态机进入死锁状态。这使得 0xF0 无法用作该模块的唤醒密钥。

**权变措施：**

不发送 0xF0 唤醒信号，而是：

发送一个带有未使用标识符的 LIN 报头 (例如保留标识符 0x3E 或 LIN 网络中未使用的任何其他标识符)

该报头中包含的间隔场将：

- 作为网络上其他 LIN 节点的有效唤醒命令
- 成功清除该模块内部的 LIN POWERDOWN 位

## i2502

### **PBIST 存储器组与目标存储器之间的映射不正确。**

#### 详细信息：

PBIST ROM 配置存在映射错误，其按存储体编号跨内核对 TCM 存储器进行分组，而不是将同一内核的两个存储体归为一组。这导致 PBIST 无法独立测试单个内核的 TCMB，因为它总是同时测试两个内核中的相同存储体。

#### 权变措施：

通过 MMR (存储器映射寄存器) 写入操作直接编写 PBIST 算法，从而绕过错误的 PBIST ROM 配置。此方法可按内核正确划分存储体组，并避免跨内核存储器损坏问题。对于 TCMA 和 TCMB 存储器测试，请参照下述特定的软件序列，用正确的存储器映射参数来配置 PBIST 模块。

TCMA 存储器测试序列：

1. 初始化 PBIST 模块：
  - 向 MSS\_CTRL\_TOP\_PBIST\_KEY\_RST 依次写入 0x05、0xA5，使 PBIST 退出复位状态并使能时钟。
2. 配置 PBIST 激活：
  - 向 PBIST\_PACT 寄存器写入 0x1。
3. 初始化循环寄存器：
  - 向所有循环寄存器 (L0、L1、L2、L3) 写入 0x0。
4. 设置覆盖模式：
  - 向 PBIST\_OVR 写入 0x9 以启用算法和 RINFO 覆盖。
5. 启用配置访问：
  - 向 PBIST\_DLR 写入 0x10。
6. 配置 PBIST 算法 (行进式扰动增量算法)：
  - 向 RF 寄存器 (RF0L 至 RF15L - 低 16 个寄存器) 写入算法代码。
  - 向 RF 寄存器 (RF0U 至 RF15U - 高 16 个寄存器) 写入算法代码。
7. 配置待测存储器：
  - 向 PBIST\_CMS (片选掩码) 写入 0x0。
  - 写入特定的 RAMT 值 (例如，对于 R5SS0 Core0 TCMA，写入 0x3691271C)。
  - 使用 CA3、CA2、CA1、CA0 (列地址寄存器) 配置地址范围。
  - 配置 CL3、CL2、CL1、CL0 (列限制寄存器)。
  - 配置 CI3、CI2、CI1、CI0 (列增量寄存器)。
  - 向 PBIST\_CS 写入片选值。
  - 向 PBIST\_PC 写入 0x1 以启动配置。

TCMB 存储器测试序列 (第一至第七步的初始化步骤与 TCMA 相同)：

1. 初始化 PBIST 模块。
2. 配置 PBIST 激活。
3. 初始化循环寄存器。
4. 设置覆盖模式。
5. 启用配置访问。
6. 配置 PBIST 算法 (行进式扰动增量算法)。
7. 配置待测存储器。
8. 启用 VIM 中断：
  - 将 0x80 写入 VIM0\_INTR\_EN\_SET\_2。
9. 配置和测试首个存储器实例：
  - 用特定的 RAMT 值 (例如 0x3400271C) 进行配置。
  - 使用 CA、CL、CI 寄存器设置地址范围。
  - 设置片选 (CS = 0x00000027)。
  - 向 PBIST\_PC 写入 0x1 以启动测试。

**i2502 (续)*****PBIST 存储器组与目标存储器之间的映射不正确。***

10. 等待首个实例测试完成：
  - 轮询 VIM0\_STS\_2 寄存器，等待设置位 0x80。
  - 向 VIM0\_STS\_2 写入 0x80 以清除中断。
11. 配置并测试剩余的七个存储器实例：
  - 对于每个剩余实例：
    - 用下一个存储器实例的值更新 RAMT 寄存器。
    - 用相应的值更新 CS 寄存器。
    - 向 PBIST\_PC 写入 0x1 以启动测试。
    - 轮询 VIM0\_STS\_2 等待完成 ( 位 0x80 ) 。
    - 向 VIM0\_STS\_2 写入 0x80 以清除中断。

**i2503*****在闪存引导模式下，无法从闪存的冗余引导位置进行引导*****详细信息：**

在每种闪存引导模式下，AM263Px ROM 均会尝试从闪存中的以下偏移量处引导引导加载程序映像：

- 0x0000\_0000
- 0x00002\_000
- 0x00004\_000
- 0x00006\_000

在此情况下，从 0x0000\_0000 以外的任何其他位置引导都会失败。

根本原因已确定为：在 SFDP 失败回退到 1s 模式期间，读数据捕获延迟寄存器的处理不正确。

**权变措施：**

无。

## 商标

所有商标均为其各自所有者的财产。

## 4 修订历史记录

注：以前版本的页码可能与当前版本的页码不同

Changes from MAY 30, 2024 to MAY 30, 2026 (from Revision E (May 2024) to Revision F (May 2026))	Page
• 增加了公告 i2488；时钟：用于精确 50-50 占空比时钟的 PLL 配置.....	2
• 增加了公告 i2508；安全系统中 RC OSC 的使用.....	4
• 增加了公告 i2499；突发读取期间发生单错误检测时，返回给主机的数据不正确.....	21
• 增加了公告 i2500；使用 0xF0 作为唤醒密钥时 LIN 模块无法唤醒.....	22
• 新增了公告 i2502；PBIST 存储器组与目标存储器之间的映射不正确。.....	23
• 增加了公告 i2503；在闪存引导模式下，无法从闪存的冗余引导位置进行引导.....	24

## 重要通知和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、与某特定用途的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他安全、安保法规或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的相关应用。严禁以其他方式对这些资源进行复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。对于因您对这些资源的使用而对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，您将全额赔偿，TI 对此概不负责。

TI 提供的产品受 [TI 销售条款](#)、[TI 通用质量指南](#) 或 [ti.com](#) 上其他适用条款或 TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。除非德州仪器 (TI) 明确将某产品指定为定制产品或客户特定产品，否则其产品均为按确定价格收入目录的标准通用器件。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

版权所有 © 2026，德州仪器 (TI) 公司

最后更新日期：2025 年 10 月