

## Errata

**Errata AM261 Sitara™ 微控制器器件版本 1.0****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**

|                                   |           |
|-----------------------------------|-----------|
| <b>1 使用说明和公告模型</b> .....          | <b>2</b>  |
| <b>2 器件修订版本 1.0 使用说明和公告</b> ..... | <b>3</b>  |
| 2.1 器件修订版本 1.0 使用说明.....          | 3         |
| 2.2 器件修订版本 1.0 公告.....            | 3         |
| <b>3 商标</b> .....                 | <b>14</b> |
| <b>4 修订历史记录</b> .....             | <b>14</b> |

## 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 |
|----|--------------------------------|------------|
|    |                                | AM261      |
|    |                                | 1.0        |
| 时钟 | i2324 — GCM 和 GCD 状态信号之间不存在同步器 | 是          |

表 1-2. 公告汇总表

| 模块        | 说明   | 受影响的器件修订版本 |
|-----------|--|------------|
|           |  | AM261      |
|           |  | 1.0        |
| 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          |
| 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          |
| 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          |
| CONTROLSS | i2356 — CONTROLSS-ADC：如果未设置 INTxCONT (继续中断模式)，中断可能会停止  | 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          |
| 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          |
| 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          |
| 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          |
| UART      | i2310 — USART：错误地触发超时中断  | 是          |
| UART      | i2311 — USART：虚假 DMA 中断  | 是          |
| DTHE      | i2428 — DTHE 中的 AES 在 GCM 加密结束时为 data_in 生成额外的 dma 请求  | 是          |
| 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          |
| USB       | i2412 — 发生 DMA 读取/写入访问错误时，USB 无法生成中断   | 是          |
| TCM       | i2411 — TCM 不支持 128 字节突发访问   | 是          |
| OSPI      | i2383 — OSPI：PHY DDR 模式不支持 2 字节地址  | 是          |
| 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          |
| OSPI      | i2351 — OSPI：直接访问控制器 (DAC) 不支持带 NAND 闪存连续读取模式  | 是          |
| OSPI      | i2189 — OSPI：控制器 PHY 调优算法  | 是          |
| CPSW      | i2440 — CPSW：主机到以太网时间戳序列 ID 问题   | 是          |
| CPSW      | i2439 — CPSW：主机到以太网时间戳精度问题   | 是          |
| ICSS      | i2433 — ICSS：读取 LSW 时，读取 64 位 IEP 计时器没有锁定 MSW 逻辑   | 是          |

## 2 器件修订版本 1.0 使用说明和公告

本节列出了此器件修订版本的使用说明和公告。

### 2.1 器件修订版本 1.0 使用说明

本节列出了适用于器件修订版本 1.0 [和早期器件修订版本]的所有使用说明。

#### ***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。

### 2.2 器件修订版本 1.0 公告

以下警告涉及功能技术规格的已知设计例外情况。警告按照添加到本文档中的顺序进行编号。一些警告编号可能会在本文档未来的修订版本中删除，因为相应设计例外已被修复或已经记录在特定于器件的数据手册或技术参考手册中。在删除一些项目时，其余警告编号不会重新排序。

#### ***i2189*** ***OSPI：控制器 PHY 调优算法***

---

**详细信息：**

启用 PHY 模块后，OSPI 控制器使用 DQS 信号对数据进行采样。但是，该模块中存在一个问题，要求这种采样必须在内部时钟定义的窗口内进行。读取操作会受到外部延迟的影响，该外部延迟会随温度变化而变化。为了保证在任何温度下都能有效读取，必须实施一种特殊调优算法来选择最稳健的 TX、RX 和读取延迟值。

**权变措施：** 有关此错误权变措施的详细说明，请参阅应用程序说明 spract2 ( 链接：<https://www.ti.com/lit/spract2> )。要在某些 PVT 条件下对数据进行采样，必须调高读取延迟字段，以改变内部时钟采样窗口。这样就可以在数据眼图内的任何位置对数据进行采样。但是，这会带来以下副作用：

1. 必须为所有读取操作启用 PHY 流水线模式。由于必须要对写入操作禁用 PHY 流水线模式，因此读取和写入操作必须分开进行。
2. 当权变措施实施后，BUSY 位的硬件轮询中断，因此必须改用软件轮询。写入必须在页边界内通过 DMA 访问进行，以防主机或闪存器件发起的中断。软件必须在页写入操作间隙轮询 BUSY 位。或者，在启用硬件轮询时，写入可以在非 PHY 模式下执行。
3. STIG 读取必须填充额外的字节，并且接收到的数据必须右移。

#### ***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

**i2310 (续)****USART: 错误地清除/触发超时中断**

如果超时中断被错误地清除：

- 清除正常，因为 FIFO 中的挂起数据将重触发超时中断

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 TIMEOUTL 寄存器中设置超时计数器的高值
- 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：

- 清除正常，因为下一个周期性事件将重触发超时中断
- 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：

- 这将导致 DMA 被软件驱动程序关闭
- 正常，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 )。

**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。

**i2351**

**OSPI : 直接访问控制器 (DAC) 不支持带 NAND 闪存的连续读取模式**

---

**详细信息 :**

OSPI 直接访问控制器 (DAC) 不支持带 NAND 闪存的连续读取模式, 因为 OSPI 控制器可以在发送到 OSPI 控制器的内部 DMA 总线请求之间, 将向闪存存储器发送的 CSn 信号置为无效 (根据设计意图)。

发生此问题的原因是, 一些 OSPI/QSPI NAND 闪存存储器提供的“连续读取”模式要求片选输入在整个突发事务中持续置为有效状态。

SoC 内部 DMA 控制器和其他启动器限制用于 1023 B 或更小的事务, 并且仲裁/排队可以在各种 DMA 控制器内部进行, 也可以在任何 DMA 控制器与 OSPI 外设之间的互连中进行。这会导致到 OSPI 控制器的总线请求延迟, 进而导致外部 CSn 信号被置为无效。

NOR 闪存存储器不受 CSn 置为无效的影响, 连续读取模式正常运行。

**权变措施 :**

软件可以使用页面/缓冲读取模式来访问 NAND 闪存。

**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) :

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。

**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;//清除 INT1 标志
```

```
if (1 == AdcaRegs.ADCINTOVF.bit.ADCINT1)//ADCINT 溢出
```

```
{
```

```
AdcaRegs.ADCINTFLGCLR.bit.ADCINT1 = 1; //再次清除 INT1
```

```
//如果应用程序将忽略 ADCINTOVF 条件
```

```
//然后通过向 ADCINTOVFCLR 写入 1 来清除此处的标志。
```

```
//如果有 ADCINTOVF 处理例程，则插入任意一个
```

```
//该代码并清除此处的 ADCINTOVF 标志，或者不清除
```

```
//此处设置 ADCINTOVF，使外部例程检测到
```

```
//条件。
```

```
// AdcaRegs.ADCINTOVFCLR.bit.ADCINT1 = 1;//清除 OVF
```

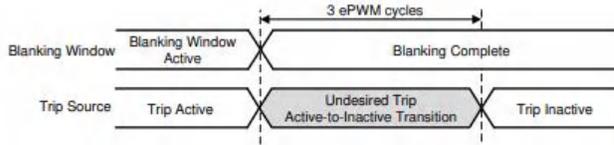
**i2357**

**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**

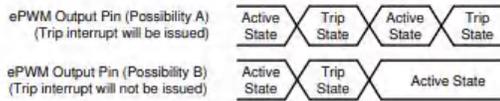
**详细信息 :**

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。



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端子系统 (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。

**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。

**i2383**

***OSPI : PHY DDR 模式不支持 2 字节地址***

---

**详细信息 :**

当 OSPI 控制器配置为在 PHY DDR 模式中进行 2 字节寻址时，内部状态机将发送的地址字节数失配为值 1 (而不是 2)。这会导致状态机在地址阶段锁定，使 PHY DDR 模式不可运行。

使用任何 Tap 模式或 PHY SDR 模式时，不会发生此问题。在 PHY DDR 模式下使用 4 字节寻址时，也不会出现此问题。

**权变措施 :**

对于具有可编程地址字节设置的兼容 OSPI 存储器，请将闪存上所需的地址字节数从 2 设置为 4。这可能涉及发送一条特定命令以更改地址字节和/或在闪存上写入配置寄存器。完成后，将在控制器设置中发送的地址字节数从 2 更新为 4。

对于仅支持 2 字节寻址且无法重新编程的兼容 OSPI 存储器，PHY DDR 模式将与该存储器不兼容。替代模式包括：

PHY SDR 模式

TAP (无 PHY) DDR 模式

TAP (无 PHY) SDR 模式

**i2411**

***不支持 128 字节突发访问***

---

**详细信息 :**

由于互连中的问题，任何主器件对 TCM 的 128 字节突发访问不能按预期工作，从而导致数据过时。

**权变措施**

请勿使用 128 字节突发访问。请在复位后由软件禁用所有主器件。

**i2412**

***发生 DMA 读取/写入访问错误时，USB 无法生成中断***

---

**详细信息 :**

如果 USB 的 DMA R/W 访问在 VBUSM 上看到读取/写入错误响应，则 USB 没有生成错误事件/中断的机制。如果在访问存储器时出现地址/MPU 错误，从而导致总线挂起，则可能会发生这种情况。

**权变措施 :**

1. 软件需要为 TRB 和数据编程正确的地址。
2. 使用 USB 传输超时并读取以下记录总线错误的寄存器

GSTS.BUSERRADDRVLD (主机/器件模式)

USBSTS.HSE (主机模式)

**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。

**i2427 (续)**
**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**

为更加清晰起见，图 2-1 列出了问题适用（例如 1、2 和 3）和不适用于（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<br>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<br>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<br>Full Write Addr C<br>Full Write Addr D ← last write prior to partial write<br>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<br>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<br>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<br>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<br>Note: N=0   | Read ← Clears bad internal state  | No spurious writes will all command combinations in subsequent cycles     |

**图 2-1.**
**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。

**i2428 (续)**

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

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) : 0xFFFFFFFF0 (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\*/

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
qbgt skip_iep_read_errata. r2, r4
;re-read IEP if IEP_COUNTER_LOW >= 0xFFFF_FFFC
lbc0 &r2, c26, 0x10, 8
skip_iep_read_errata:
```

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

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;
    }
}
    
```

**i2439**
**CPSW : 主机到以太网时间戳精度问题**

详细信息:

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。

**i2440**
**CPSW : 主机到以太网时间戳序列 ID 问题**

详细信息:

对于启用了时间戳的主机到以太网数据包, 某些情况可能会导致与时间戳关联的序列 ID 不正确。时间戳正确, 但序列 ID 不正确。主机将使用序列 ID 将数据包与时间戳相关联。此问题不会影响 PTP 消息。

权变措施:

对于主机到以太网, 带有时间戳的数据包应该被禁用。

i2479

**GPIO61 复位引脚配置的 OSPI 引导问题**

**详细信息：**

在 OSPI 引导模式期间 ( 包括会导致 OSPI 引导模式的回退模式 ) ， ROM 将引脚 GPIO61 配置为 OSPI0\_RESET\_OUT0。如果该引脚连接到 OSPI 闪存的复位引脚，则可能会由于 OSPI 控制器中的复位信号管理问题而发生引导失败。

在 OSPI 引导模式期间，ROM 代码：

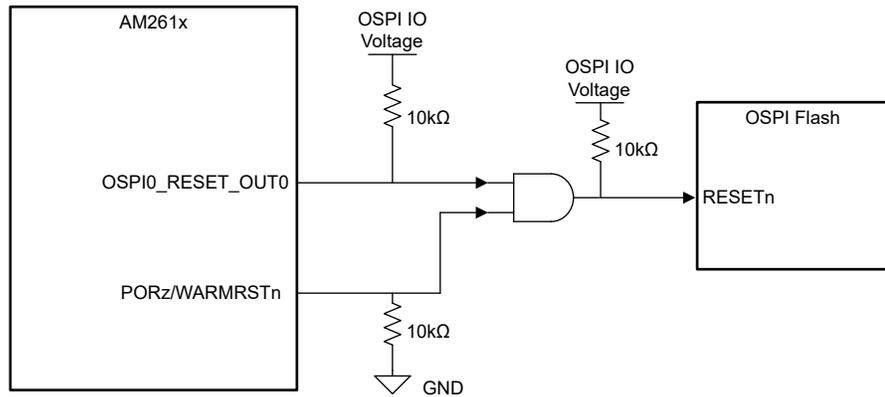
1. 在 OSPI0\_RESET\_OUT0 多路复用器模式下配置 GPIO61
2. 正确地将 RESET 信号置为有效 ( 驱动低电平 )
3. 由于 OSPI 控制器配置不正确，无法将 RESET 信号取消置为有效 ( GPIO61 引脚保持驱动为低电平 )

如果 GPIO61 连接到闪存存储器复位引脚，则闪存器件保持复位状态。这会阻止正确的引导序列完成。该问题会影响将 GPIO61 用于 OSPI 闪存复位控制的所有封装类型。

**权变措施：**

选项 1：请勿使用 GPIO61 引脚复位闪存。使用 PORz/WARMRSTn 在下电上电时复位闪存器件，并为 OSPI0\_RESET\_OUT0 使用另一个 GPIO，并将该信号通过与门与 PORz/WARMRSTn 传递。有关使用任何 OSPI0\_RESET\_OUT0 引脚与 PORz/WARMRSTn 的示例，请参阅 AM261x OSPI 复位。

选项 2：选通 GPIO61 以防止在引导期间传播到 RESET 逻辑，然后在闪存器件的下电上电期间将该与 PORz/WARMRSTn 的信号用作复位。使用选通 GPIO61 引脚与 PORz/WARMRSTn 进行 AM261x OSPI 复位中显示了一个示例实现。



**图 2-2. 使用任何 OSPI0\_RESET\_OUT0 引脚与 PORz/WARMRSTn 进行的 AM261x OSPI 复位**

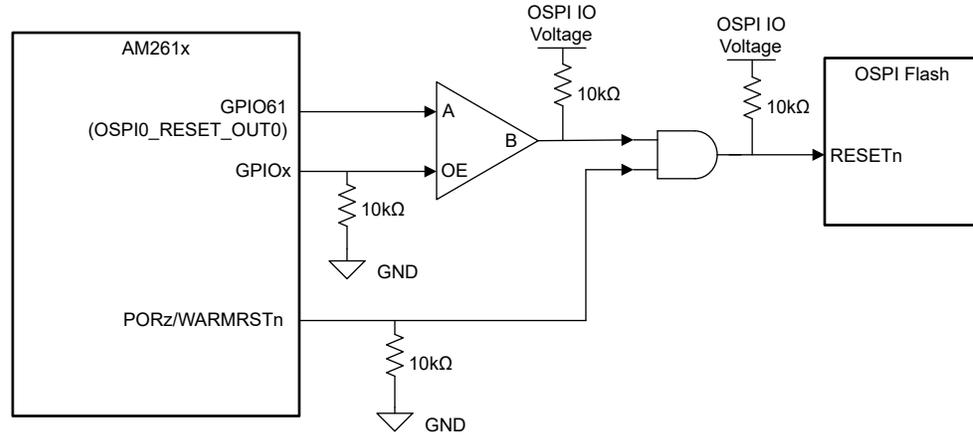


图 2-3. 使用选通 GPIO61 引脚与 PORz/WARMRSTn 进行的 AM261x OSPI 复位

详细信息：

在 OSPI 引导模式期间（包括会导致 OSPI 引导模式的回退模式），ROM 将引脚 GPIO61 配置为 OSPI0\_RESET\_OUT0。如果该引脚连接到 OSPI 闪存的复位引脚，则可能会由于 OSPI 控制器中的复位信号管理问题而发生引导失败。

在 OSPI 引导模式期间，ROM 代码：

1. 在 OSPI0\_RESET\_OUT0 多路复用器模式下配置 GPIO61
2. 正确地将 RESET 信号置为有效（驱动低电平）
3. 由于 OSPI 控制器配置不正确，无法将 RESET 信号取消置为有效（GPIO61 引脚保持驱动为低电平）

如果 GPIO61 连接到闪存存储器复位引脚，则闪存器件保持复位状态。这会阻止正确的引导序列完成。该问题会影响将 GPIO61 用于 OSPI 闪存复位控制的所有封装类型。此外，GPIO61 将保持低电平，直到用户应用程序重新配置它。

权变措施：

选项 1：无。请勿使用 GPIO61 引脚。

选项 2：如果需要使用此引脚，用户应用代码可以重新配置此引脚。但是如果应用对于干扰敏感，则应通过在外部电路中添加适当的滤波来确保干扰不会影响连接的外设。

3 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 NOVEMBER 7, 2024 to APRIL 30, 2025 (from Revision \* (November 2024) to Revision A (April 2025))

|   | Page |
|---|------|
| • 增加了公告 i2479：GPIO61 复位引脚配置的 OSPI 引导问题.....                     | 13   |
| • 增加了公告 i2480：OSPI 引导期间 GPIO61/OSPI0_RESET_OUT0 上的 1 μs 干扰..... | 14   |

## 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、某特定用途方面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功能安全、信息安全、监管或其他要求。

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。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/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。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265  
版权所有 © 2025，德州仪器 (TI) 公司