

## Errata

**F29H85x、F29P58x 和 F29P32x 实时 MCU 器件勘误表 ( 器件修订版本 C、B、A、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>  |
| 1.1 使用说明汇总表                | 2         |
| 1.2 公告汇总表                  | 2         |
| <b>2 命名规则、封装编号法和修订版本标识</b> | <b>4</b>  |
| 2.1 器件和开发支持工具命名规则          | 4         |
| 2.2 支持的器件                  | 4         |
| 2.3 封装编号法和修订版本标识           | 5         |
| <b>3 器件修订版本 B 使用说明和公告</b>  | <b>7</b>  |
| 3.1 器件修订版本 B 使用说明          | 7         |
| 3.2 器件修订版本 B 公告            | 7         |
| <b>4 器件修订版本 A 使用说明和公告</b>  | <b>25</b> |
| 4.1 器件修订版本 A 使用说明          | 25        |
| 4.2 器件修订版本 A 公告            | 25        |
| <b>5 器件修订版本 0 使用说明和公告</b>  | <b>31</b> |
| 5.1 器件修订版本 0 使用说明          | 31        |
| 5.2 器件修订版本 0 公告            | 31        |
| <b>6 文档支持</b>              | <b>33</b> |
| <b>7 商标</b>                | <b>33</b> |
| <b>8 修订历史记录</b>            | <b>33</b> |

## 插图清单

|                                     |    |
|-------------------------------------|----|
| 图 2-1. ZEX 封装的封装编号法                 | 5  |
| 图 2-2. PTS 封装的封装编号法                 | 5  |
| 图 2-3. RFS 封装的封装编号法                 | 5  |
| 图 2-4. PZS 封装的封装编号法                 | 6  |
| 图 3-1. 具有 AGPIO 和 AIO 模拟引脚类型的模拟子系统图 | 10 |
| 图 3-2. 不良跳闸事件和消隐窗口过期                | 11 |
| 图 3-3. 可能会产生不良的 ePWM 输出             | 11 |
| 图 3-4. 同步 ISR 加载方法示例                | 15 |
| 图 3-5. 错误的上电序列导致卡在复位位置              | 22 |
| 图 3-6. 具有复位释放的正确上电序列                | 22 |

## 表格清单

|                      |    |
|----------------------|----|
| 表 1-1. 使用说明汇总表       | 2  |
| 表 1-2. 公告汇总表         | 2  |
| 表 2-1. 版本标识          | 6  |
| 表 3-1. 特定模拟输入引脚的用例组合 | 10 |
| 表 5-1. 时钟源选项         | 32 |

## 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。

### 1.1 使用说明汇总表

表 1-1. 使用说明汇总表

| 编号      | 标题                          | 受影响的器件修订版本 |   |   |   |
|---------|-----------------------------|------------|---|---|---|
|         |                             | 0          | A | B | C |
| 节 4.1.1 | 安全性：在器件修订版本 B 器件上编程的新 TI 密钥 | 是          | 是 | 否 | 否 |

### 1.2 公告汇总表

表 1-2. 公告汇总表

| 模块          | 说明                                                                                                 | 受影响的器件修订版本 |   |   |   |
|-------------|----------------------------------------------------------------------------------------------------|------------|---|---|---|
|             |                                                                                                    | 0          | A | B | C |
| ADC         | ADC：如果未设置 INTxCONT（继续中断模式），中断可能会停止                                                                 | 是          | 是 | 是 | 是 |
| ADC         | ADC：使用内部 VREF 的 16 位转换模式下的峰值代码跳变                                                                   | 是          | 是 | 是 | 是 |
| CAN         | 器件修订版本 0、A 不支持 CAN 引导模式                                                                            | 是          | 是 | 否 | 否 |
| CMPSS       | CMPSS：如果比较器输入引脚具有 AGPIO 功能并且 ADC 正在对输入引脚进行采样，则可能会发生 CMPSS 干扰                                       | 是          | 是 | 是 | 是 |
| MCAN        | 当从配置有相同报文 ID 的专用 Tx 缓冲区发送时，报文顺序反转                                                                  | 是          | 是 | 是 | 是 |
| ePWM        | 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 | 是 | 是 |
| ePWM        | ePWM：当 PERCLKDIVSEL.EPWMCLKDIV = 1 时，ePWM TZFRC 和 TZCLR 事件可能会丢失                                    | 是          | 是 | 是 | 是 |
| ePWM        | ePWM：跳闸区的 ESM 源不受支持                                                                                | 是          | 是 | 是 | 是 |
| ePWM        | ePWM：ePWM 单次/CBC 跳闸事件 DCxEVty.force 不设置跳闸条件                                                        | 是          | 是 | 是 | 是 |
| ePWM        | ePWM：对于在单次加载模式下使用全局加载的 ePWM，当对 GLDCTL2.OSHTLD 寄存器进行写入发生在全局加载事件的 3 个 TBCLK 内时，全局加载寄存器可能会延迟          | 是          | 是 | 是 | 是 |
| 闪存          | 闪存：启用闪存预取后的独立 CPU1/CPU3 复位可能会导致 CPU1/CPU3 出现 NMI                                                   | 是          | 是 | 否 | 否 |
| FOTA        | FOTA：具有加密功能的安全 FOTA 不起作用                                                                           | 是          | 是 | 否 | 否 |
| HRPWM       | HRPWM：HRPWM 的高分辨率周期影子寄存器到活动寄存器的加载会在每个零事件发生时执行，即便周期的影子寄存器到活动寄存器的加载模式已设置为仅在 SYNC 时加载                 | 是          | 是 | 是 | 是 |
| HSM         | HSM：HSM ROM 代码不会引导大小超过 191KB 的 HSMRT 映像                                                            | 是          | 是 | 否 | 否 |
| LIN         | LIN：LIN 无法使用 0xF0 唤醒密钥唤醒                                                                           | 是          | 是 | 是 | 是 |
| MCD         | MCD：启用 PLL (PLLCLKEN = 1) 后，应该禁用时钟丢失检测                                                             | 是          | 是 | 是 | 是 |
| MEMSS       | MEMSS：RAM 的数据线缓冲区 (DLB) 会导致数据一致性问题                                                                 | 是          | 是 | 是 | 是 |
| ROM         | ROM：默认情况下，GPIO4 通过 ROM 代码配置为 ERRORSTS 并置为高电平                                                       | 是          | 否 | 否 | 否 |
| SDFM        | 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 | 是 | 是 |
| SDFM        | 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 | 是 | 是 |
| SDFM        | 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 | 是 | 是 |
| C29 CPU 子系统 | C29 CPU 子系统：HS-FS 器件的 C29 CPU 子系统中未触发的 DTHE 中断和 DMA 事件                                             | 是          | 是 | 否 | 否 |
| 系统          | 系统：当 VDD 电压在 VDDIO 之前斜升时，器件复位仍然置位                                                                  | 是          | 是 | 是 | 是 |
| 系统          | 系统：CPU 进入故障状态后，流水线中待处理的未对齐读取会阻止 NMI 向量的获取                                                          | 是          | 是 | 否 | 否 |
| 系统          | 系统：复位将导致 CPU 卡在 BOOTROM 的 NMI 处理程序中                                                                | 是          | 是 | 否 | 否 |
| 系统          | 系统：如果 SYSCLKDIVSEL.PLLSYSCLKDIV = 0，则发出器件复位信号 (XRSn) 可能会导致意外故障                                     | 是          | 是 | 是 | 是 |
| 系统          | 系统：器件内部复位可能会导致器件卡在复位循环中                                                                            | 是          | 是 | 否 | 否 |
| UART        | UART：UART FIFO 在连续调试器读取时被清除                                                                        | 是          | 是 | 是 | 是 |

表 1-2. 公告汇总表 ( 续 )

| 模块     | 说明                                                  | 受影响的器件修订版本 |   |   |   |
|--------|-----------------------------------------------------|------------|---|---|---|
|        |                                                     | 0          | A | B | C |
| VSSOSC | <a href="#">VSSOSC</a> : 从相邻引脚 X1 或 X2 的耦合可能会阻止正确引导 | 是          | 否 | 否 | 否 |

## 2 命名规则、封装编号法和修订版本标识

### 2.1 器件和开发支持工具命名规则

德州仪器 (TI) 为其支持工具推荐使用三种可能的前缀指示符中的两个：TMDX 和 TMDS。这些前缀代表了产品开发的发展阶段，即从工程原型 (TMDX) 直到完全合格的生产工具 (TMDS)。

器件开发演变流程：

**X** 试验器件不一定代表最终器件的电气规范标准，并且可能不使用生产组装流程。

**P** 原型器件不一定是最终器件模型，并且不一定符合最终电气标准规范。

**无** 完全合格的芯片模型的生产版本。

支持工具开发演变流程：

**TMDX** 还未经德州仪器 (TI) 完整内部质量测试的开发支持产品。

**TMDS** 完全合格的开发支持产品。

X 和 P 器件和 TMDX 开发支持工具在供货时附带如下免责条款：

“开发的产品用于内部评估用途。”

生产器件和 TMDS 开发支持工具已进行完全特性描述，并且器件的质量和可靠性已经完全论证。TI 的标准保修证书适用。

预测显示原型器件 (X 或者 P) 的故障率大于标准生产器件。由于这些器件的预期最终使用故障率仍未确定，故德州仪器 (TI) 建议请勿将这些器件用于任何生产系统。请仅使用合格的生产器件。

### 2.2 支持的器件

本文档支持以下器件：

- [F29H850TU](#)
- [F29H859TU-Q1](#)
- F29H859TM-Q1
- F29H850DU
- F29H859DU-Q1
- F29H850DM
- F29H859DM-Q1
- F29P589DU-Q1
- F29P580DM
- F29P589DM-Q1
- F29P329SM-Q1

## 2.3 封装编号法和修订版本标识

图 2-1、图 2-2、图 2-3 和图 2-4 展示了封装编号法。表 2-1 列出了器件修订版本代码。

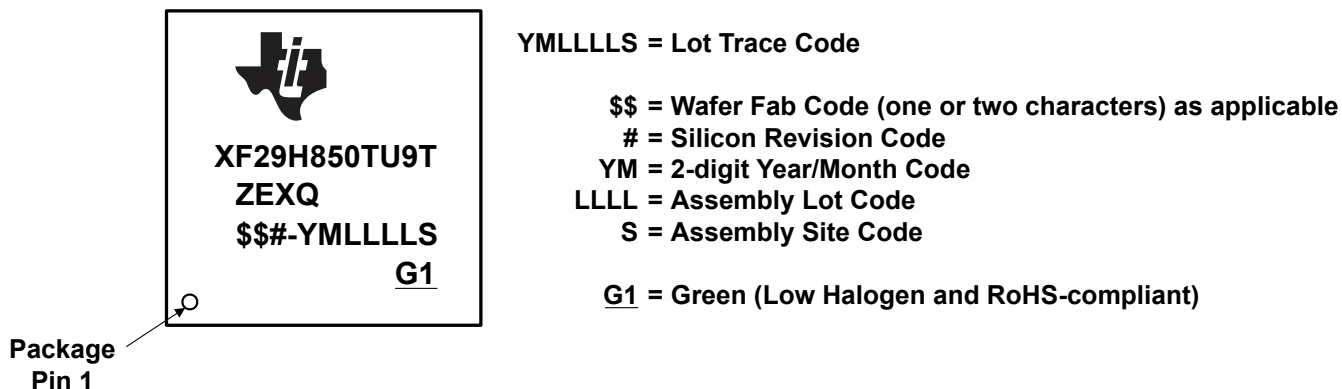


图 2-1. ZEX 封装的封装编号法

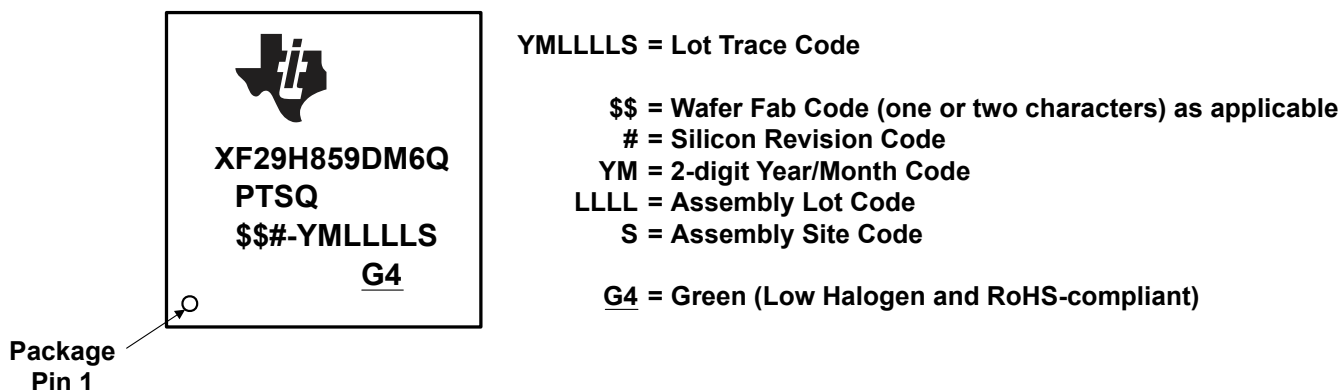


图 2-2. PTS 封装的封装编号法

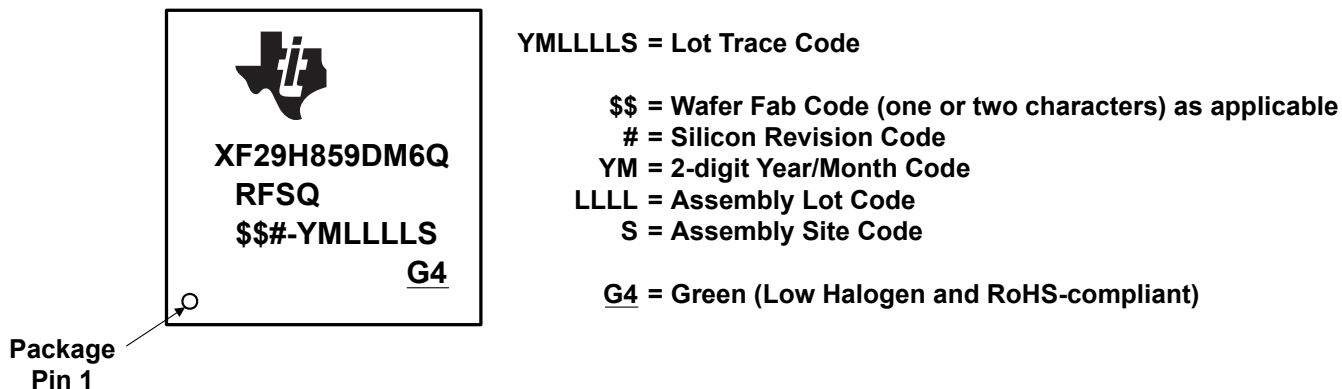


图 2-3. RFS 封装的封装编号法

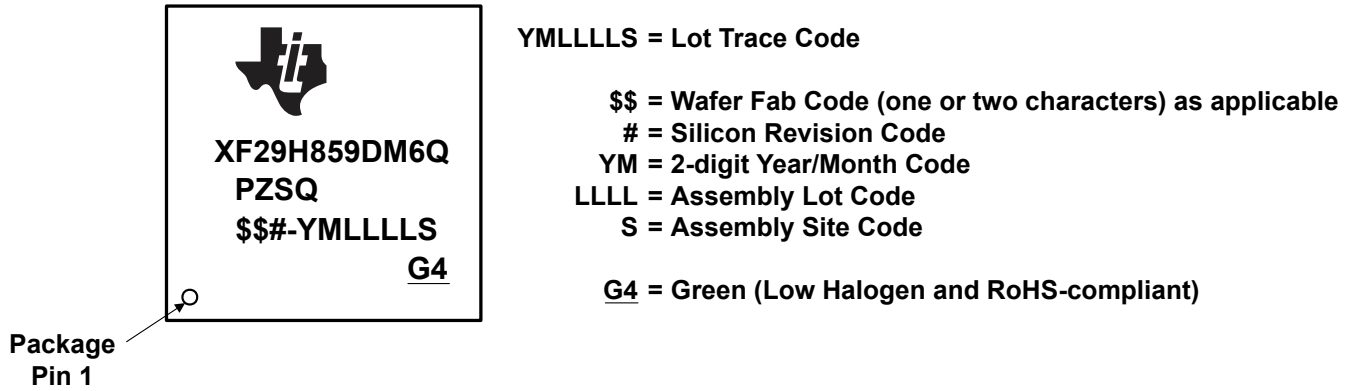


图 2-4. PZS 封装的封装编号法

表 2-1. 版本标识

| 器件修订版本代码 | 器件修订版本 | REVID <sup>(1)</sup><br>地址：0x3018 0028 | 注释 <sup>(2)</sup> |
|----------|--------|----------------------------------------|-------------------|
| 空白       | 0      | 0x0000 0001                            | 该器件版本现已预量产。       |
| A        | A      | 不适用 <sup>(3)</sup>                     | 该器件版本现已预量产。       |
| B        | B      | 0x0000 0003                            | 该器件现已量产           |
| C        | C      | 0x0000 0004                            | 该器件现已量产           |

- (1) 器件修订版本 ID
- (2) 有关可订购器件型号，请参阅 [F29H85x](#)、[F29P58x](#) 和 [F29P32x](#) 实时微控制器数据表中的“封装信息”表。
- (3) 请参阅封装编号法图中的器件修订版本代码。

### 3 器件修订版本 B 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。

#### 3.1 器件修订版本 B 使用说明

本节列出了适用于器件修订版本 B 和早期器件修订版本的所有使用说明。

#### 3.2 器件修订版本 B 公告

本节列出了适用于器件修订版本 B 和早期器件修订版本的所有公告。

##### 公告

**ADC：如果未设置 *INTxCONT*（继续中断模式），中断可能会停止**

##### 受影响版本

0、A、B、C

##### 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。

##### 应变方法

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
}
```

##### 公告

**ADC：使用内部 *VREF* 的 16 位转换模式下的峰值代码跳变**

##### 受影响版本

0、A、B、C

##### 详细信息

在使用内部 *VREF* 模式的 16 位 ADC 转换中，20 秒内可能至少发生一次代码跳变。代码跳变幅度可以近似为 110 个代码 \* (*VIN/VREF*)。当采样信号振幅 *VIN* 接近 *VREFHI* 电平

## 公告 (续)

**ADC：使用内部 VREF 的 16 位转换模式下的峰值代码跳变**

时，会观察到峰值代码跳变。在外部 VREF 模式下未观察到此代码跳变。在外部 VREF 模式下未观察到代码跳变。在 12 位转换模式下也不会观察到该情况。

## 解决方法

如果选择 16 位 ADC 内部 VREF 模式，那么在应用条件允许时，可执行过采样转换或采用平均滤波，以大幅降低代码跳变带来的影响。另一个选择是使用外部 VREF。

|       |                                                                                                                                        |
|-------|----------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | 当从配置有相同报文 ID 的专用 Tx 缓冲区发送时，报文顺序反转                                                                                                      |
| 受影响版本 | 0、A、B、C                                                                                                                                |
| 详细信息  | 多个 Tx 缓冲区配置有相同的报文 ID。按升序顺序提出发送这些 Tx 缓冲区的请求，每个 Tx 请求之间会有一定延迟。根据各个 Tx 请求之间的延迟，Tx 缓冲区可能不会按 Tx 缓冲区编号的预期升序顺序进行发送。                           |
| 应变方法  | <p>首先，将具有相同报文 ID 的一组 Tx 报文写入报文 RAM。然后，通过对 <b>TXBAR</b> 的一次写访问，同时请求发送所有这些报文。</p> <p>使用 Tx FIFO 而不是专用 Tx 缓冲区来按特定顺序发送多条具有相同报文 ID 的报文。</p> |

## 公告

**CMPSS**：如果比较器输入引脚具有 **AGPIO** 功能并且 **ADC** 正在对输入引脚进行采样，则可能会发生 **CMPSS** 干扰

## 受影响版本

0、A、B、C

## 详细信息

需要特别注意特定模拟输入引脚的用例组合，如表 3-1 所示。如此表所示，对于 **CMPSS** 输入、**ADC** 采样和 **AGPIO** 的组合，需要使用特殊注意事项或解决方法。

表 3-1. 特定模拟输入引脚的用例组合

| 特定模拟引脚上使用的功能 | 使用的元件  |   |             |   |   |
|--------------|--------|---|-------------|---|---|
| CMPSS 比较器输入  | 是      | - | 是           | - | 是 |
| ADC 采样       | 是      | 是 | -           | 是 | 是 |
| AGPIO 模拟引脚类型 | 是      | 是 | 是           | - | - |
| AIO 模拟引脚类型   | -      | - | -           | 是 | 是 |
| 结果           | 需要权变措施 |   | 无需特殊分析或权变措施 |   |   |

**AGPIO** 模拟引脚路径包含一个额外的  $53\Omega$  串联开关。这会创建一个由 **ADC** 和 **CMPSS** 比较器共享的低电容隔离式节点，如图 3-1 所示。当 **ADC** 对通道进行采样时，该节点可能会受到干扰（取决于 **ADC** 采样保持电容器上先前存储的电压），这种干扰可能会导致高达  $50\text{ns}$  的错误 **CMPSS** 事件。为了适应这种潜在的干扰，可以实施以下权变措施。

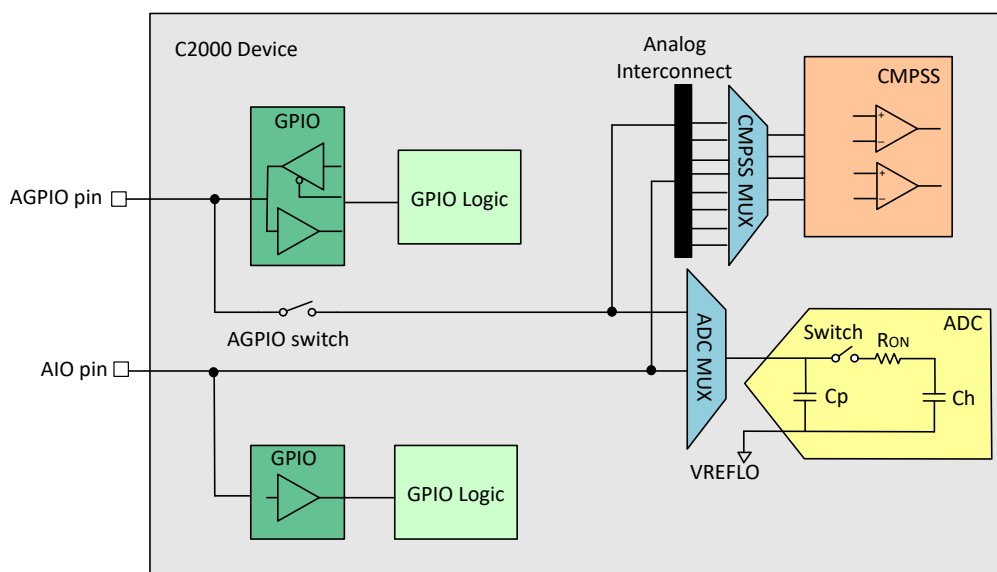


图 3-1. 具有 **AGPIO** 和 **AIO** 模拟引脚类型的模拟子系统图

## 解决方法

- 对于同时需要 **ADC** 和 **CMPSS** 的模拟通道，使用不同的引脚（即 **AIO** 引脚类型）。
- 使用设置为  $50\text{ns}$  或更大的 **CMPSS** 数字滤波器，从而将滤除临时干扰。
- 预处理 **ADC** 的采样保持电容器，从而将使干扰不会导致误跳闸。例如，在读取受影响的通道之前，立即对 **ADC** 上不同通道的  $3.3\text{V}$  连接执行虚拟读取，从而使干扰为正，远离误跳闸。如果误跳闸极性反转，则将使用  $0\text{V}$  信号的相反虚拟读取。

## 公告

**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

## 受影响版本

0、A、B、C

## 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。

图 3-2 展示了可能会导致不良 ePWM 输出的时间段。

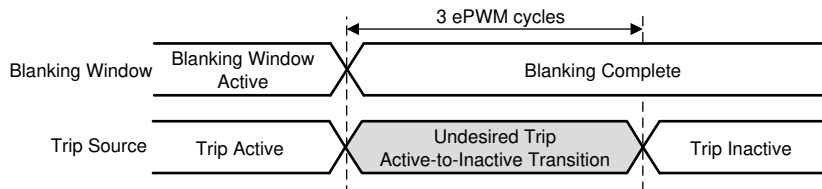


图 3-2. 不良跳闸事件和消隐窗口过期

图 3-3 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。

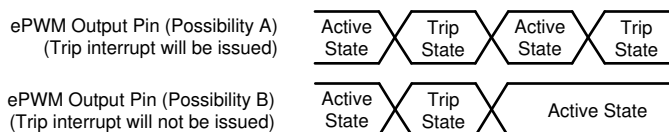


图 3-3. 可能会产生不良的 ePWM 输出

## 权变措施

扩展或减小消隐窗口，以避免任何不良的跳闸操作。

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>ePWM</b> : 当 <b>PERCLKDIVSEL.EPWMCLKDIV = 1</b> 时, <b>ePWM TZFRC</b> 和 <b>TZCLR</b> 事件可能会丢失                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 受影响版本 | 0、A、B、C                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 详细信息  | <p>TZFRC 位用于软件强制跳闸事件, 而 TZCLR 位用于清除跳闸区事件。在具有 EPWMCLKDIV 的器件上, 如果 PERCLKDIVSEL.EPWMCLKDIV 编程为 1, TZFRC 和 TZCLR 写入可能会丢失, 并且输出不受影响。该位默认编程为 1 ( EPWMCLK 为 PLLSYSCLK/2 )。</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 权变措施  | <ol style="list-style-type: none"> <li>1. 配置 <b>EPWMCLK = PLLSYSCLK (PERCLKDIVSEL.EPWMCLKDIV = 0)</b>。</li> <li>2. 如果用户必须配置 <b>EPWMCLK = PLLSYSCLK/2 (PERCLKDIVSEL.EPWMCLKDIV = 1)</b>, 则使用以下 driverlib 软件序列, 选择 EPWMXBAR 的一个保留的多路复用器输入, 将其用于该跳闸。 <p>PWMXBAR → 数字比较 → 跳闸区</p> <ol style="list-style-type: none"> <li>1. 为 EPWMxA 上的数字比较输出 A 事件 1/2 操作以及 EPWMxB 上的数字比较输出 B 事件 1/2 操作配置跳闸区操作。 <ul style="list-style-type: none"> <li>• EPWM_setTripZoneAction()</li> </ul> </li> <li>2. 配置 TRIPIN1-15 的输入信号或 TRIPIN1-15 的“或”组合逻辑。 <ul style="list-style-type: none"> <li>• EPWM_selectDigitalCompareTripInput()</li> </ul> </li> <li>3. 为 DCAEVT1/2 和 DCBEVT1/2 配置数字比较条件。 <ul style="list-style-type: none"> <li>• EPWM_setTripZoneDigitalCompareEventCondition()</li> </ul> </li> <li>4. 将数字比较子模块的 PWMXBAR 输入配置为保留。 <ul style="list-style-type: none"> <li>• XBAR_selectEpwmXbarInputSource()</li> </ul> </li> </ol> </li> </ol> <p><b>应用代码</b></p> <p>要触发 PWM, 可以使用 XBAR_invertOutputSignal() 反转 PWMXBAR 状态。</p> |

公告

**ePWM : 跳闸区的 ESM 源不受支持**

---

受影响版本

0、A、B、C

详细信息

来自 ESM 的 ESM\_GEN\_EVENT 可能无法正确跳闸 ePWM，因此不应使用。

以下路径不应用于触发 PWM：

ESM 子系统 (SYS ESM) → ESM\_GEN\_EVENT → PWM XBAR → ePWM 数字比较 →  
ePWM 跳闸区域

权变措施

使用错误源从相应的 ESM CPU 实例生成 NMI 中断。在 NMI 中断内，使用下面的 Driverlib 函数发出软件写入以强制 PWM 输出跳闸。

EPWM\_forceTripZoneEvent()

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>ePWM : ePWM 单次/CBC 跳闸事件 DCxEVTy.force 不设置跳闸条件</b>                                                                                                                                                                                                                                                                                                                                                                                      |
| 受影响版本 | 0、A、B、C                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 详细信息  | <p>DCxEVTy.force 信号可用作单次跳闸事件和逐周期跳闸事件的输入。设置 DCxCTL[EVT1/2FRCSYNCSEL] = 1 ( 通过异步路径传递 DCxEVT 信号 ) 不会正确设置单次或逐周期跳闸事件条件, 并且跳闸事件丢失。</p>                                                                                                                                                                                                                                                                                                          |
| 权变措施  | <p>对于具有 DCxCTL.EVT1/2FRCSYNCSEL 和 DCxCTL.EVT1/2LATSEL 的器件, 存在两个选项。否则, “Sync Path” 是唯一可用于单次或逐周期跳闸事件的选项。</p> <p><b>异步路径 ( 通过异步路径传递 DCxEVT 信号 )</b></p> <p>要通过异步路径锁存单次或逐周期跳闸条件, 必须进行以下配置:</p> <ol style="list-style-type: none"><li>1. DCxCTL.EVT1/2FRCSYNCSEL = 1</li><li>2. DCxCTL.EVT1/2LATSEL = 1</li></ol> <p><b>同步路径 ( 通过 SYNC 路径传递 DCxEVT 信号 )</b></p> <ol style="list-style-type: none"><li>1. DCxCTL.EVT1/2FRCSYNCSEL = 0</li></ol> |

## 公告

**ePWM**：对于在单次加载模式下使用全局加载的 ePWM，当对 GLDCTL2.OSHTLD 寄存器进行写入发生在全局加载事件的 3 个 TBCLK 内时，全局加载寄存器可能会延迟

## 受影响版本

0、A、B、C

## 详细信息

当对 GLDCTL2.OSHTLD 寄存器位的写入在全局加载事件 ( 由 GLDCTL.GLDMODE 配置 ) 的 3 个 TBCLK 内发生时，寄存器的全局加载 ( 由 GLDCFG 配置 ) 可能会延迟，由此会产生意外的波形。

## 权变措施

为了避免该问题，在配置的全局加载事件之前，对 GLDCTL2.OSHTLD 寄存器位的任何写入都需要至少 3 个 TBCLK 周期。

### 权变措施 1 ( 推荐 )：同步 ISR 到 PWM

1. 针对已知事件触发中断事件 ( 即 CMPC、CMPD 事件等 ) 生成 ISR。
2. 确保在全局加载事件和 ISR 写入 GLDCTL2.OSHTLD 时至少保持 3 个 TBCTR 周期的间隔。
  - a. 写入 OSHTLD 位时，请禁用中断，确保这是不可中断的代码。

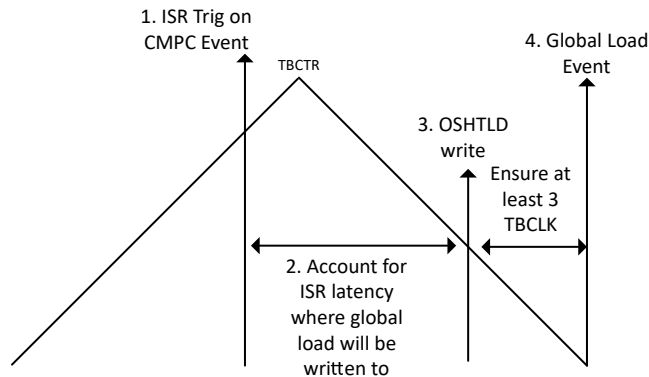


图 3-4. 同步 ISR 加载方法示例

### 权变措施 2：PWM 寄存器的异步更新

1. 在写入 GLDCTL2.OSHTLD 寄存器之前，从启用了全局加载的 PWM 实例 TBCTR 读取。
2. 确保在全局加载事件与写入 GLDCTL2.OSHTLD 之间至少保持 3 个周期。
  - a. 通过禁用中断，确保从 TBCTR 读取代码并写入 OSHTLD 位是不可中断的代码。

|       |                                                                                                                                              |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>HRPWM</b> : HRPWM 的高分辨率周期影子寄存器到活动寄存器的加载会在每个零事件发生时执行，即便周期的影子寄存器到活动寄存器的加载模式已设置为仅在 <b>SYNC</b> 时加载                                           |
| 受影响版本 | 0、A、B、C                                                                                                                                      |
| 详细信息  | 当启用高分辨率周期 (HRPCTL[HRPE] = 1)，并且影子寄存器到活动寄存器的加载配置为在发生 <b>SYNC</b> 事件时加载 (TBCTL2[PRDLDSYNC] = 0x1 或 0x2) 时，加载会在每个零事件发生，而非在同步事件时进行。              |
| 权变措施  | <p>当使用高分辨率周期时，<b>SYNC</b> 事件会被忽略，且影子寄存器到活动寄存器的加载会在每个零事件发生。</p> <p>如果禁用高分辨率周期，则不会忽略 <b>SYNC</b> 事件，且在发生 <b>SYNC</b> 事件时会发生影子寄存器到活动寄存器的加载。</p> |

|       |                                                                                                        |
|-------|--------------------------------------------------------------------------------------------------------|
| 公告    | <b>LIN: LIN 无法使用 0xF0 唤醒密钥唤醒</b>                                                                       |
| 受影响版本 | 0、A、B、C                                                                                                |
| 详细信息  | 此问题仅在使用 LIN 作为命令器时适用。如果 LIN 尝试发送 0xF0 作为唤醒密钥，内部状态机会卡住且无法恢复，从而阻止后续的传输。                                  |
| 权变措施  | 发送未使用的 LIN 接头而非 0xF0 唤醒信号。可以使用保留标识符 0x3E 或其他任何未使用的标识符。接头内的中断域会作为有效 LIN 唤醒命令作用于其他 LIN 节点，并禁用内部 LIN 断电位。 |

|       |                                                                                                                                                                                                                                        |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>MCD : 启用 PLL (PLLCLKEN = 1) 后, 应该禁用时钟丢失检测</b>                                                                                                                                                                                       |
| 受影响版本 | 0、A、B、C                                                                                                                                                                                                                                |
| 详细信息  | <p>PLL 具有跛行模式功能, 可提供慢速 PLLRAWCLK 输出 (即使缺少其输入 OSCCLK)。</p> <p>另外, 当检测到缺少 OSCCLK 输入时, 时钟丢失检测 (MCD) 电路会强制将系统时钟源切换到 INTOSC1。当两个时钟源 (PLLRAWCLK 和 INTOSC1) 都仍然有效时, 无法确保在这些系统时钟源之间切换的 MCD 多路复用器无干扰。在极少数情况下, 这可能会导致在时钟丢失故障事件期间出现不可预测的器件行为。</p> |
| 应变方法  | <p>当系统使用 PLL (PLLCLKEN = 1) 时, 可通过写入 MCDCCR.MCLKOFF = 1 来禁用 MCD。</p> <p>可对双时钟比较器 (DCC) 电路进行配置, 以快速检测 SYSCLK 频率是否由于时钟丢失事件而降至低于所需频率, 导致进入跛行模式。</p> <p>当系统以 PLL 旁路模式 (PLLCLKEN = 0) 运行时, 仍可使用 MCD 电路来检测时钟丢失事件并将时钟源切换到 INTOSC1。</p>        |

|       |                                                                                            |
|-------|--------------------------------------------------------------------------------------------|
| 公告    | <b>MEMSS : RAM 的数据线缓冲区 (DLB) 会导致数据一致性问题</b>                                                |
| 受影响版本 | 0、A、B、C                                                                                    |
| 详细信息  | 当启用数据线缓冲区 (DLB) (默认情况下) 并且两个 CPU 对同一 RAM 地址位置同时执行读取/写入操作时, 在某些情况下, 读取操作可能会接收到过时的数据, 而非新数据。 |
| 权变措施  | 如果 RAM 块在多个 CPU 之间共享, 用户应使用 MEM_DLB_CONFIG 寄存器中的配置位禁用 DLB。                                 |

|       |                                                                                                                                                                                                                                                                                                                            |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>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</b>                                                                                                                                                                                                                                                                    |
| 受影响版本 | 0、A、B、C                                                                                                                                                                                                                                                                                                                    |
| 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、ePWM X-BAR 事件和 GPIO 输出 X-BAR 事件。                                                                                                                                                                                      |
| 权变措施  | <p>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 中断或 X-BAR 事件 (ePWM X-BAR/GPIO 输出 X-BAR 事件) :</p> <ol style="list-style-type: none"> <li>1. 禁用比较器滤波器。</li> <li>2. 至少延迟比较器滤波器的延时 + 3 个 SD-Cx 时钟周期。</li> <li>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。</li> <li>4. 至少延迟比较器滤波器的延时 + 5 个 SD-Cx 时钟周期。</li> <li>5. 启用比较器滤波器。</li> </ol> |
| 公告    | <b>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</b>                                                                                                                                                                                                                                                                        |
| 受影响版本 | 0、A、B、C                                                                                                                                                                                                                                                                                                                    |
| 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 中断和 DMA 触发器。                                                                                                                                                                                                                          |
| 权变措施  | <p>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 :</p> <ol style="list-style-type: none"> <li>1. 禁用数据滤波器。</li> <li>2. 至少延迟数据滤波器的延时 + 3 个 SD-Cx 时钟周期。</li> <li>3. 更改数据滤波器设置, 如滤波器类型和 DOSR。</li> <li>4. 至少延迟数据滤波器的延时 + 5 个 SD-Cx 时钟周期。</li> <li>5. 启用数据滤波器。</li> </ol>                                                  |

|       |                                                                                                                                                                                      |
|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>SDFM</b> ：在三个 SD 调制器时钟周期内对 <b>SDCPARMx</b> 寄存器位字段 <b>CEVT1SEL</b> 、 <b>CEVT2SEL</b> 和 <b>HZEN</b> 进行两次背对背写入可能会损坏 <b>SDFM</b> 状态机，从而导致虚假比较器事件                                      |
| 受影响版本 | 0、A、B、C                                                                                                                                                                              |
| 详细信息  | 在三个 SD 调制器时钟周期内对 <b>SDCPARMx</b> 寄存器位字段 <b>CEVT1SEL</b> 、 <b>CEVT2SEL</b> 和 <b>HZEN</b> 进行背对背写入可能会损坏 <b>SDFM</b> 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、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 <b>SDCPARMx</b> 寄存器位字段。                                                                                                                          |

## 公告

系统：当 VDD 电压在 VDDIO 之前斜升时，器件复位仍然置位

## 受影响版本

0、A、B、C

## 详细信息

当 VDD 电源电压在 VDDIO 电源之前或同时斜升时，器件 XRSn 复位信号可以保持低电平（复位位置为有效）状态。因此，器件无法引导。

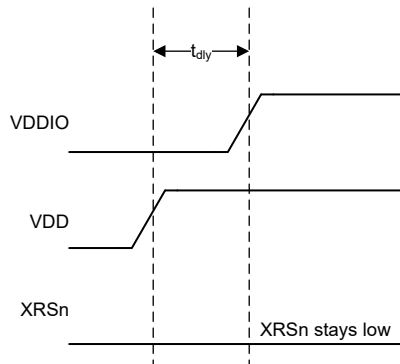


图 3-5. 错误的上电序列导致卡在复位位置

## 解决方法

确保 VDDIO 电源完全斜升，然后再斜升 VDD 电源电压。要查看  $V_{DD33} - V_{DD12}$  延迟参数，请参阅“电源管理模块运行条件”表

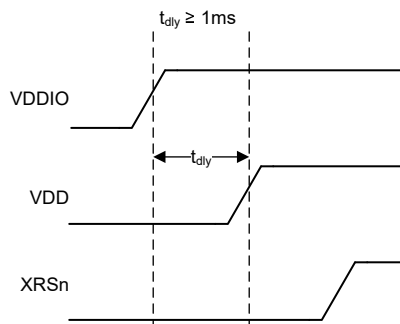


图 3-6. 具有复位释放的正确上电序列

|       |                                                                                                                           |
|-------|---------------------------------------------------------------------------------------------------------------------------|
| 公告    | 系统：如果 <code>SYSCLKDIVSEL.PLLSYSCLKDIV = 0</code> ，则发出器件复位信号 ( <code>XRSn</code> ) 可能会导致意外故障                               |
| 受影响版本 | 0、A、B、C                                                                                                                   |
| 详细信息  | 当 <code>SYSCLKDIVSEL.PLLSYSCLKDIV = 0</code> ( 1 分频 ) 时，发出器件复位信号 ( <code>XRSn</code> — 内部或外部 ) 可能会导致器件进入意外故障状态。           |
| 权变措施  | 请勿使用 <code>SYSCLKDIVSEL.PLLSYSCLKDIV = 0</code> ( 1 分频 )。请改用 <code>SYSCLKDIVSEL.PLLSYSCLKDIV = 1</code> ( 2 分频 ) 或更大的分频值。 |

|       |                                                                           |
|-------|---------------------------------------------------------------------------|
| 公告    | <b>UART : UART FIFO 在连续调试器读取时被清除</b>                                      |
| 受影响版本 | 0、A、B、C                                                                   |
| 详细信息  | UART IP 以相同的方式处理调试器和 CPU 读取。因此，从 CCS 连续读取 UART_DR 寄存器时，FIFO 在代码实际读取之前被清空。 |
| 权变措施  | 在 UART 数据传输期间，请勿使内存浏览器保持打开状态。                                             |

## 4 器件修订版本 A 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。

### 4.1 器修订版本 A 使用说明

在后续的器件修订版本中，可参阅器件修订版本适用的使用说明。欲了解更多详细信息，请参阅[器件修订版本 B 的使用说明](#)。

#### 4.1.1 安全性：在器件修订版本 B 器件上编程的新 TI 密钥

受影响的版本：0、A

已在器件修订版本 B 器件上对新的 FEK ( 文件加密密钥 ) 值进行编程。客户需要更改其密钥并使用 OTP KW 版本和 TIFS ( TI 基础安全 ) 封装版本 1.2.1。器件修订版本 B 器件仅适用于所述的封装版本。

### 4.2 器件修订版本 A 公告

较晚的器件修订版本中已收录适用器件修订版本的公告。有关更多详细信息，请参阅[器件修订版本 B 公告](#)。

|       |                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>闪存：启用闪存预取后的独立 CPU1/CPU3 复位可能会导致 CPU1/CPU3 出现 NMI</b>                                                                                                                                                                                                                                                                                                                                                            |
| 受影响版本 | 0、A                                                                                                                                                                                                                                                                                                                                                                                                                |
| 详细信息  | <p>如果启用了闪存预取，向 CPU1 或 CPU3 发出的独立复位 ( 例如，来自 CCS 的调试复位 ) 可能会由于不可纠正的 ECC 错误而导致 CPU NMI。以下是独立 CPU 复位的源。</p> <p>CPU1：</p> <ol style="list-style-type: none"> <li>1. 调试复位</li> <li>2. HSM → CPU1.RSn</li> </ol> <p>CPU3：</p> <ol style="list-style-type: none"> <li>1. 调试复位</li> <li>2. HSM → CPU3.RSn</li> <li>3. CPU3 WD 复位</li> <li>4. CPU3 NMIWD 复位</li> <li>5. SSU_CPU3_CFG_REGS → CPU_RST_CTRL.SW_SYSRN</li> </ol> |
| 权变措施  | <p>在向 CPU 发出独立复位之前禁用闪存预取 (FRIx_INTF_CTRL.PREFETCH_EN = 0)。</p> <p>CPU1 → FRI1_INTF_CTRL.PREFETCH_EN = 0</p> <p>CPU3 → FRI3_INTF_CTRL.PREFETCH_EN = 0</p> <p>如果启用了 ZONE1，则可以从 CPU1.LINK2 和 CPU1/3 调试器写入 FRIx_INTF_CTRL.PREFETCH_EN。</p>                                                                                                                                                                             |

|       |                                        |
|-------|----------------------------------------|
| 公告    | <b>FOTA : 具有加密功能的安全 FOTA 不起作用</b>      |
| 受影响版本 | 0、A                                    |
| 详细信息  | 引导序列中的漏洞会阻止对空中下载固件 (FOTA) 映像应用加密和密钥派生。 |
| 权变措施  | 无。该问题已在器件修订版本 B 中修复。                   |

|       |                                                      |
|-------|------------------------------------------------------|
| 公告    | <b>HSM : HSM ROM 代码不会引导大小超过 191KB 的 HSMRT 映像</b>     |
| 受影响版本 | 0、A                                                  |
| 详细信息  | 在闪存引导模式下，硬件安全模块 (HSM) ROM 不会引导大小超过 191KB 的 HSMRT 映像。 |
| 权变措施  | 将 HSMRT 映像大小保持在 191KB 以下。                            |

|       |                                                                                                                                                    |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>C29 CPU 子系统 : HS-FS 器件的 C29 CPU 子系统中未触发的中断和 DMA 事件</b>                                                                                          |
| 受影响版本 | 0、A                                                                                                                                                |
| 详细信息  | 在高安全性 - 现场安全 (HS-FS) 器件生命周期状态下，可以使用引导证书扩展选项将加密引擎自动分配给 C29 CPU 子系统。如果在证书中配置了此选项，则会将引擎映射到 C29 CPU，但相应的中断信号和 DMA 事件不会路由到 C29 CPU，因此在运行 C29 应用程序时不会触发。 |
| 权变措施  | 要检测由加密加速器引擎生成的事件，请轮询相应的中断或 DMA 状态寄存器。                                                                                                              |

|       |                                                                                                                                                                                                                                                                                                                                                           |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | 系统：CPU 进入故障状态后，流水线中待处理的未对齐读取会阻止 NMI 向量的获取                                                                                                                                                                                                                                                                                                                 |
| 受影响版本 | 0、A                                                                                                                                                                                                                                                                                                                                                       |
| 详细信息  | 当由未对齐读取导致三个或更多连续的 C29 CPU 故障时，NMI 处理程序无法执行。当 CPU 流水线中有两个以上的故障时，CPU 不会按预期获取 NMI 向量。                                                                                                                                                                                                                                                                        |
| 权变措施  | <p>使用 ERAD-SEC 计数器：</p> <ol style="list-style-type: none"><li>1. 选择 ESM_GEN_EVENT 作为 EPWMXBAR 的输入。</li><li>2. 在启动/停止模式下配置 ERAD-SEC1 计数器：启动事件作为 EPWMXBAR 事件，停止事件作为 SEC1 事件本身。该计数器将对 SYSCLK 周期进行计数。</li><li>3. 配置 ERAD-SEC 基准寄存器，以生成匹配事件并在计数为 50 时触发 NMI ( SEC_CNTL 寄存器中的 INT_EN 和 NMI_EN 位 )。</li><li>4. 配置 ESM CPU1 以针对 ERAD CPU1_NMI 事件生成 NMI。</li></ol> |

|       |                                                                                                                                                                                                                                                           |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>系统：复位将导致 CPU 卡在 BOOTROM 的 NMI 处理程序中</b>                                                                                                                                                                                                                |
| 受影响版本 | 0、A                                                                                                                                                                                                                                                       |
| 详细信息  | <p>在 F29P32x 器件上，ADCE 不存在，但 BOOTROM 代码访问 ADCE 寄存器；因此，CPU 会收到访问故障消息并卡在 BOOTROM 的 NMI 处理程序中，从而可能导致闪存编程错误等问题。</p>                                                                                                                                            |
| 权变措施  | <ul style="list-style-type: none"> <li>此问题不会在加电时发生；因此，用户应在 CCS 的“Flash Settings”中取消选中“Reset target before flash programming/operations”选项。</li> <li>在应用中禁用 WD 以避免任何器件复位。此外，请勿从应用发出任何复位命令。</li> <li>如果需要复位器件，请对器件进行电源断电重启。</li> </ul>                      |
| 公告    | <b>系统：器件内部复位可能会导致器件卡在复位循环中</b>                                                                                                                                                                                                                            |
| 受影响版本 | 0、A                                                                                                                                                                                                                                                       |
| 详细信息  | <p>在修订版本 0/A 器件上，当发生可触发 XRSn 电平翻转的器件内部复位时，HSM 时钟分频器 (HSMCLKDIVSEL, HSMCLKDIV) 会先复位为 0 (/1)，随后其余时钟逻辑（如 PLL 设置）才完成复位。这会让 HSM 子系统短暂以 200MHz 系统时钟运行数个时钟周期，该频率超出 HSM 规定的 100MHz 工作频率。进而导致 HSM 进入未定义状态，器件最终因看门狗复位而卡在复位循环中。</p> <p>此问题已在修订版本 B 和后续修订版本中得以修复。</p> |
| 应变方法  | 无。出现此错误状况时需对器件断电重启。建议改用修订版本 C 裸片开展应用开发。                                                                                                                                                                                                                   |
| 公告    | <b>器件修订版本 0、A 不支持 CAN 引导模式。</b>                                                                                                                                                                                                                           |
| 受影响版本 | 0、A                                                                                                                                                                                                                                                       |
| 详细信息  | 在早期器件修订版本 0 和 A 中，未在 BootROM 中实现 CAN 引导模式选项。                                                                                                                                                                                                              |
| 权变措施  | 无。如果需要 CAN 引导模式，请使用器件修订版本 B 或更高版本。                                                                                                                                                                                                                        |

## 5 器件修订版本 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。

### 5.1 器件修订版本 0 使用说明

在后续的器件修订版本中，可参阅器件修订版本适用的使用说明。欲了解更多详细信息，请参阅[器件修订版本 B 的使用说明](#)。

### 5.2 器件修订版本 0 公告

较晚的器件修订版本中已收录适用器件修订版本的公告。有关更多详细信息，请参阅器件[修订版本 A 公告](#)和[器件修订版本 B 公告](#)。

|       |                                                                                                                                                                                                                                                                                                                                                                                                          |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 公告    | <b>ROM：默认情况下，GPIO4 通过 ROM 代码配置为 ERRORSTS 并置为高电平</b>                                                                                                                                                                                                                                                                                                                                                      |
| 受影响版本 | 0                                                                                                                                                                                                                                                                                                                                                                                                        |
| 详细信息  | <p>ROM 代码根据 SECCFG 字段设置，在器件复位 (XRSn) 后为 ERRORSTS 功能配置 GPIO 引脚。默认情况下，ROM 代码将 GPIO4 引脚配置为 ERRORSTS 引脚。ERRORSTS (GPIO4) 引脚将由错误信令模块 (ESM) 控制，并默认被置动为高电平 (无错误)。如果 GPIO4 引脚用于驱动关键系统功能，这种高电平状态可能会导致电路板问题。例如，如果 GPIO4 用于 EPWM3_A 功能，则高功率 FET 可能会无意中导通，从而导致电路板损坏。</p>                                                                                                                                             |
| 权变措施  | <ol style="list-style-type: none"> <li>1. 避免将 GPIO4 用于电路板上的关键系统功能 (例如，驱动高功率 FET)。</li> <li>2. 如果 GPIO4 必须用于关键系统功能，请通过适当配置 SECCFG 字段，为 ERRORSTS 选择不同的 GPIO。请参阅 SDK 中的 empty_driverlib_project_secure 示例，了解如何更改 ERRORSTS 引脚多路复用 (pinmux) 选项。</li> </ol> <p>由于 ROM 代码仅在器件复位 (XRSn) 时为 ERRORSTS 功能配置 GPIO 引脚，但也会通过调试器复位来清除 ERRORSTS 引脚多路复用 (pinmux) 配置，因此用户需要发出完整的器件复位 (XRSn) 以重新运行 ERRORSTS 的 ROM 代码配置。</p> |

**公告****VSSOSC : 来自相邻引脚 X1 或 X2 的耦合可能会妨碍正常引导****受影响版本**

0

**详细信息**

对于器件修订版本 0 器件，VSSOSC 噪声会导致 INTOSC2 干扰，当 INTOSC2 为器件的时钟源时，这会使器件进入故障状态。在引导、闪存编程以及应用配置的任何其他时间期间，INTOSC2 用作器件的主时钟。

相邻 X1 或 X2 引脚上的快速边沿速率会在 VSSOSC 引脚上引起噪声，对于器件修订版 0 器件，必须避免这种情况。器件修订版本 0 之后的器件没有此灵敏度。

**权变措施**

在器件修订版本 0 器件上，请勿在 X1 或 X2 引脚上使用快速边沿。例如，不要使用单端晶体来驱动 X1。

**表 5-1. 时钟源选项**

| 应用时钟源            | 器件修订版本 0                    | 器件修订版本 A | 器件修订版本 B |
|------------------|-----------------------------|----------|----------|
| X1 - X2 晶体       | 优选                          | 可接受      | 可接受      |
| 来自外部振荡器的 X1 单端输入 | 避免                          | 可接受      | 可接受      |
| INTOSC2          | 如果更宽的 INTOSC 频率精度可接受，则此方案可行 |          |          |

## 6 文档支持

欲获得器件专用数据表和相关文档，请访问 TI 网站：<https://www.ti.com>。

欲了解有关 F29H85x、F29P58x 和 F29P32x 器件的更多信息，请参阅以下文档：

- [F29H85x、F29P58x 和 F29P32x 实时微控制器数据表](#)
- [F29H85x 和 F29P58x 实时微控制器技术参考手册](#)

## 7 商标

所有商标均为其各自所有者的财产。

## 8 修订历史记录

### Changes from JULY 30, 2026 to AUGUST 30, 2026 (from Revision E (July 2026) to Revision F (August 2026))

|                                                                                                                                             | Page |
|---------------------------------------------------------------------------------------------------------------------------------------------|------|
| • 添加了 <a href="#">ADC</a> ：使用内部 VREF 的 16 位转换模式下的峰值代码跳变公告.....                                                                              | 7    |
| • 添加了 <a href="#">CMPSS</a> ：如果比较器输入引脚具有 <a href="#">AGPIO</a> 功能并且 <a href="#">ADC</a> 正在对输入引脚进行采样，则可能会发生 <a href="#">CMPSS</a> 干扰公告。..... | 10   |
| • <a href="#">系统</a> ：当 <a href="#">VDD</a> 电压在 <a href="#">VDDIO</a> 之前斜升时，器件复位仍然置位公告：更新了解决方法。.....                                        | 22   |

## 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