

Errata

TMS320F280013x 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

1 使用说明和公告模型	3
1.1 使用说明汇总表.....	3
1.2 公告汇总表.....	3
2 命名规则、封装编号法和修订版本标识	4
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
3 器件修订版本 C 使用说明和公告	7
3.1 器件修订版本 C 使用说明.....	7
3.2 器件修订版本 C 公告.....	9
4 器件修订版本 B 使用说明和公告	29
4.1 器件修订版本 B 使用说明.....	29
4.2 器件修订版本 B 公告.....	29
5 器件修订版本 A 使用说明和公告	30
5.1 器件修订版本 A 使用说明.....	30
5.2 器件修订版本 A 公告.....	30
6 器件修订版本 0 使用说明和公告	31
6.1 器件修订版本 0 使用说明.....	31
6.2 器件修订版本 0 公告.....	31
7 文档支持	32
8 商标	32
9 修订历史记录	32

插图清单

图 2-1. PM 封装的封装编号法.....	5
图 2-2. PT 封装的封装编号法.....	5
图 2-3. RGZ 封装的封装编号法.....	5
图 2-4. RHB 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.....	13
图 3-2. 不良跳闸事件和消隐窗口过期.....	15
图 3-3. 可能会产生不良的 ePWM 输出.....	15
图 3-4. 流水线中没有停滞时的问题流水线图.....	18
图 3-5. 指令 11 的 E3 时隙中存在停滞时的问题流水线图.....	19
图 3-6. 带解决方法的流水线图.....	20

表格清单

表 1-1. 使用说明汇总表.....	3
表 1-2. 公告汇总表.....	3
表 2-1. 版本标识.....	6

表 3-1. ADCCTL2 寄存器.....	10
表 3-2. 特定模拟输入引脚的用例组合.....	13
表 3-3. C2000 作为带有标准模式主机的目标发送器时的数据上升时间要求.....	22
表 3-4. 公共总线电容 (C_b) 的上拉电阻 (R_p) 值.....	23
表 3-5. 受公告影响的存储器.....	24
表 3-6. OTP 修订版本号位置.....	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。

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 3.1.1	PIE：背对背 PIEACK 写入和手动 CPU 中断屏蔽清除之后的伪波嵌套中断	是	是	是	是
节 3.1.2	使用嵌套中断时要小心	是	是	是	是
节 3.1.3	安全性：主要的防御层是构建芯片安全边界，从启用 JTAGLOCK 和零引脚引导至闪存功能开始	是	是	是	是

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 ADCCLK 小数分频器时 ADC 性能下降	是	是	是	是
BOR	BOR：2.45V 至 3.0V 之间的 VDDIO 可产生多个 XRSn 脉冲	是	是	是	是
CMPSS	CMPSS：在某些情况下，COMPxLATCH 可能无法正确清除	是	是	是	是
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
DCAN	在 DCAN FIFO 模式期间，接收到的消息可能以无序状态显示在 FIFO 缓冲器中	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：在消隐窗口后的前 3 个周期内，消隐窗口不会过滤跳闸事件	是	是	是	是
eQEP	eQEP：索引期间方向变化时位置计数器复位错误	是	是	是	是
闪存	闪存：不生成单个位 ECC 错误中断	是	是	是	是
闪存	闪存 API：C2000Ware 4.03.00.00 中量产版本不可用	是	是	是	是
FPU	FPU：FPU 至 CPU 寄存器移动操作之前是任何 FPU 2p 操作	是	是	是	是
GPIO	GPIO19/X1：若未按照推荐的工作条件而施加电压，可能会导致器件出错	是	是	是	是
I2C	I2C：目标发送器模式、标准模式 SDA 时序限制	是	是	是	是
存储器	存储器：在有效存储器之外进行预取	是	是	是	是
MPOST	MPOST：在某些早期物料上不会执行存储器开机自检	是	是	是	是
系统	系统：多次连续写入 CLKSRCTL1 可能会导致系统死机	是	是	是	是
PLL	PLL 基准时钟丢失检测：丢失时钟标志可能被错误地激活	是	是	是	是
看门狗	看门狗：WDKEY 寄存器不受 EALLOW 保护	是	是	是	是

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

为了标示产品开发周期所处的阶段，TI 为所有 DSP 器件和支持工具的器件型号分配了前缀。每个 DSP 商用产品系列成员都具有以下三个前缀之一：TMX、TMP 或 TMS (例如，**TMS320F2800137**)。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 (TMX 和 TMDX) 直到完全合格的生产器件和工具 (TMS 和 TMDS)。

器件开发演变流程：

TMX 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。

TMP 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。

TMS 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。

TMX 和 TMP 器件和 TMDX 开发支持工具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。

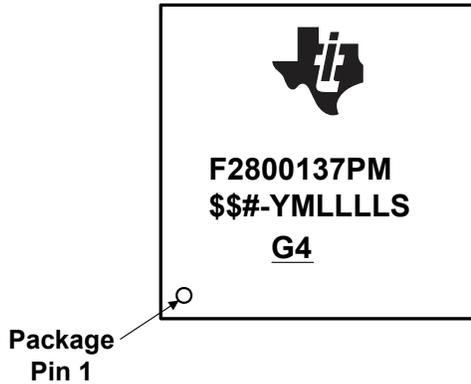
2.2 支持的器件

本文档支持以下器件：

- [TMS320F2800137](#)
- [TMS320F2800135](#)
- [TMS320F2800133](#)
- [TMS320F2800132](#)

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。

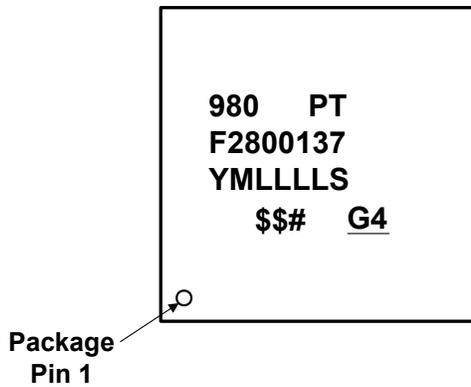


YMLLLLLS = Lot Trace Code

YM = 2-digit Year/Month Code
LLLL = Assembly Lot Code
S = Assembly Site Code
\$\$ = Wafer Fab Code (one or two characters) as applicable
= Silicon Revision Code

G4 = Green (Low Halogen and RoHS-compliant)

图 2-1. PM 封装的封装编号法

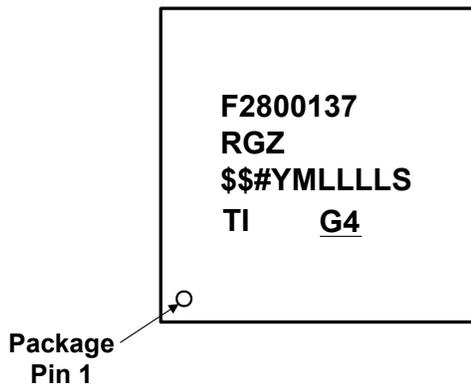


YMLLLLLS = Lot Trace Code

980 = TI EIA Code
YM = 2-digit Year/Month Code
LLLL = Assembly Lot Code
S = Assembly Site Code
\$\$ = Wafer Fab Code (one or two characters) as applicable
= Silicon Revision Code

G4 = Green (Low Halogen and RoHS-compliant)

图 2-2. PT 封装的封装编号法



YMLLLLLS = Lot Trace Code

YM = 2-digit Year/Month Code
LLLL = Assembly Lot Code
S = Assembly Site Code
\$\$ = Wafer Fab Code (one or two characters) as applicable
= Silicon Revision Code

G4 = Green (Low Halogen and RoHS-compliant)

图 2-3. RGZ 封装的封装编号法

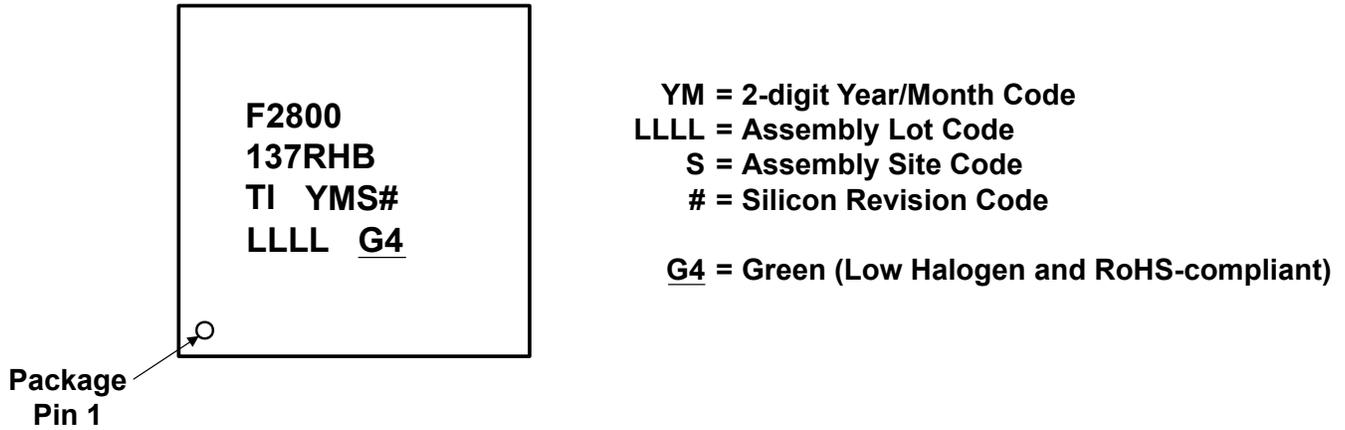


图 2-4. RHB 封装的封装编号法

表 2-1. 版本标识

器件修订版本代码	器件修订版本	REVID ⁽¹⁾ 地址：0x5D00C	注释 ⁽²⁾
空白	0	0x0000 0001	该器件修订版本以 TMX 形式提供。
A	A	0x0000 0002	该器件修订版本以 TMX 形式提供。
B	B	0x0000 0003	该器件修订版本以 TMS 形式提供。修订版 B 和 C 在功能上是等效的。
C	C	0x0000 0004	该器件修订版本以 TMS 形式提供。修订版 B 和 C 在功能上是等效的。

(1) 器件修订版本 ID

(2) 有关可订购器件型号，请参阅 [TMS320F280013x 实时微控制器数据表](#) 中的“封装信息”表。

3 器件修订版本 C 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 C 使用说明

本节列出了适用于器件修订版本 C 和早期器件修订版本的所有使用说明。

3.1.1 PIE : 背对背 PIEACK 写入和手动 CPU 中断屏蔽清除之后的伪波嵌套中断

受影响的版本 : 0、A、B、C

某些用于嵌套中断的代码序列允许 CPU 和 PIE 进入不一致状态, 从而触发不必要的中断。进入该状态所需的条件为:

1. PIEACK 清除后, 立即执行全局中断使能 (EINT 或 ASM (" CLRC INTM"))。
2. 嵌套中断会清除其组的一个或多个 PIEIER 位。

是否触发不必要的中断, 取决于系统中其他中断的配置和时序。在大多数应用中, 预计这种事件很罕见或根本不存在。如果发生这种情况, 不必要的中断将是嵌套中断的 PIE 组中的第一个中断、并将在嵌套中断重新启用 CPU 中断 (EINT 或 asm ("CLRC INTM")) 后触发。

权变措施: 在 PIEACK 写入和 CPU 中断使能之间添加一个 NOP (无操作)。以下显示了示例代码。

```

//Bad interrupt nesting code
PieCtrlRegs.PIEACK.all = 0xFFFF;      //Enable nesting in the PIE
EINT;                                    //Enable nesting in the CPU

//Good interrupt nesting code
PieCtrlRegs.PIEACK.all = 0xFFFF;      //Enable nesting in the PIE
asm(" NOP");                             //wait for PIEACK to exit the pipeline
EINT;                                    //Enable nesting in the CPU
  
```

3.1.2 使用嵌套中断时的注意事项

受影响的版本 : 0、A、B、C

如果用户为了使用嵌套特性而在中断处理例程 (ISR) 内使用 EINT 指令启用中断, 那么用户必须在退出 ISR 之前禁用中断。否则, CPU 可能会执行未确定的操作。

3.1.3 安全性：主要防御层能保护芯片边界，这首先是启用 JTAGLOCK 和零引脚引导至闪存功能

受影响的版本：0、A、B、C

在任何情况下都不允许未经授权的代码进入器件并执行，这是器件安全的前提条件。为此，器件提供了两项功能，担心安全性的用户应该始终启用这两项功能。

- **JTAGLOCK**

当在闪存的 **USER OTP** 区域中启用时，**JTAGLOCK** 功能会禁用对器件的资源进 **JTAG** 访问（例如调试器连接），阻止未经授权的一方使用 **JTAG** 接口将任何代码下载到器件中。启用 **JTAGLOCK** 后，用户仍然可以通过输入密码来允许授权方解锁，也可以通过将密码值全部设置为零来永久锁定。

- **零引脚引导至闪存**

内置于 **TI ROM** 中的外部引导加载程序不会对所下载的代码执行任何身份验证。在 **USER OTP** 中启用零引脚引导选项以及闪存引导模式，则在引导时会阻止运行所有基于引脚的外部引导加载程序选项（例如 **SCI**、**CAN**、并行），这是因为在基本引导 **ROM** 执行结束后，会强制使引导过程立即跳转到内部闪存。为了保障最高的安全性，可选择安全闪存启动模式。这可以在跳转到闪存代码之前通过基本引导 **ROM** 对闪存代码进行预检查。

如果 **JTAG** 被永久锁定并且启用了零引脚引导至闪存选项，则通过 **JTAG** 或内置引导加载程序与器件进行通信的编程工具将失效。如果需要能够执行固件升级，则用户必须将代码预存储在闪存中以安全地管理和执行更新。

3.2 器件修订版本 C 公告

本节列出了适用于器件修订版本 C 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 ADCCLK 小数分频器时 ADC 性能下降****受影响版本**

0、A、B、C

详细信息

已证明使用分数 SYSCLK 转 ADCCLK 分频器 (由 ADCCTL2.PRESCALE 字段控制) 会导致该器件的 ADC 性能下降。请参阅 [表 3-1](#)。

表 3-1. ADCCTL2 寄存器

性能下降			
位	字段	值	说明
3-0	预分频	0001	ADCCLK = SYSCLK/1.5
		0003	ADCCLK = SYSCLK/2.5
		...	
性能正常			
位	字段	值	说明
3-0	预分频	0000	ADCCLK = SYSCLK/1.0
		0002	ADCCLK = SYSCLK/2.0
		...	

权变措施

使用偶数预分频时钟分频器值。偶数预分频值会产生整数时钟分频，而不会影响 ADC 性能。

公告

BOR : 2.45V 至 3.0V 之间的 VDDIO 可产生多个 XRSn 脉冲

受影响版本

0、A、B、C

详细信息

当 VDDIO 电源电压介于 2.45V 和 3.0V 之间时，BOR 会生成重复的 XRSn 置为有效和置为无效。建议不要直接将 XRSn 引脚用作系统中任何其他器件的复位。

即使在 XRSn 脉冲发生时，F280013x BOR 也能有效地在内部将器件保持在已知的复位状态。在 VDDIO 供电电压升至 3.0 V 以上之前，该器件不会转移应用代码或引导加载程序，所有其他引脚都将保持其复位状态。

应变方法

1. 上电、断电和 BOR 事件期间，忽略额外的 XRSn 切换。额外的 XRSn 脉冲将不会影响 F280013x 器件本身的运行。
2. 如果 XRSn 脉冲会导致其他系统元件出现不良的系统行为，则请勿使用 XRSn 驱动其他器件。这些应用可使用外部电压监控器。
3. 对于需要在正常上电和断电期间避免这些脉冲的应用：
 - a. 上电：请遵循 [TMS320F280013x 实时微控制器](#) 数据表的建议运行条件表中的 SR_{SUPPLY} 要求；不会出现额外的 XRSn 低电平脉冲。
 - b. 断电：为避免 XRSn 在断电期间失效，在对电源进行设计时应确保 VDDIO 在 25 μs 内支持 3.0V 至 2.45V 的电压范围。如果 XRSn 上的电压上升是可以接受的，则可以计算 XRSn 上实现的 RC 电路的时间常数，以确保电压不会上升到系统指定的阈值以上。

公告	CMPSS : 在某些情况下, COMPxLATCH 可能无法正确清除
受影响版本	0、A、B、C
详细信息	<p>CMPSS 锁存路径旨在在本地锁存器 (COMPxLATCH) 中保持跳闸状态, 直到由软件 (通过 COMPSTSCLR) 或 PWMSYNC 清除。</p> <p>在信号经过数字滤波器数字化和鉴定后, COMPxLATCH 由比较器输出间接设置。比较器输出到达 COMPxLATCH 的预期最大延时可以用 CMPSS 模块时钟周期表示为:</p> $\text{LATENCY} = 1 + (1 \times \text{FILTER_PRESCALE}) + (\text{FILTER_THRESH} \times \text{FILTER_PRESCALE})$ <p>当 COMPxLATCH 由软件或 PWMSYNC 清除时, 锁存本身根据需要清除, 但 COMPxLATCH 之前的数据路径可能不会反映额外延时数的模块时钟周期的比较器输出值。如果在 COMPxLATCH 被清除时数字滤波器输出解析为逻辑 1, 则锁存将在下一个时钟周期再次设置。</p>
应变方法	<p>在清除 COMPxLATCH 之前, 让数字滤波器输出解析为逻辑 0。</p> <p>如果软件清除了 COMPxLATCH, 那么在清除锁存之前可以通过 COMPSTS 寄存器确认数字滤波器的输出状态。对于较大的延时值会产生不可容忍的延时的情况, 可以通过重新初始化数字滤波器 (通过 CTRIPxFILCTL) 来刷新滤波器 FIFO。</p> <p>如果 COMPxLATCH 被 PWMSYNC 清除, 设计用户应用程序时应使得比较器跳闸条件在 PWMSYNC 生成前能至少清除延时周期。</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 3-2 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权变措施。

表 3-2. 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 53Ω 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 50ns 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。

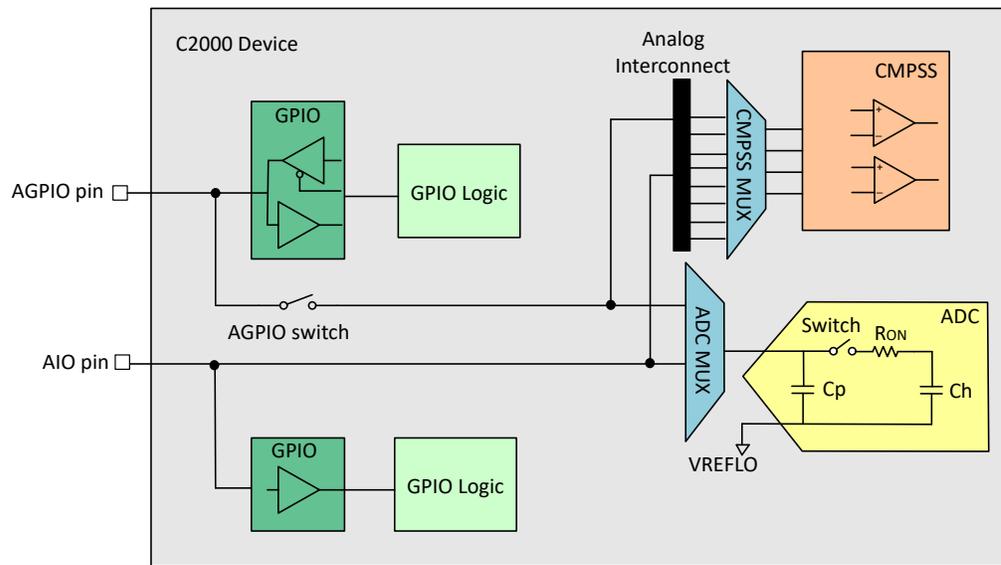


图 3-1. 具有 **AGPIO** 和 **AIO** 模拟引脚类型的模拟子系统图

应变方法

1. 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）。
2. 使用设置为 50ns 或更大的 **CMPSS** 数字滤波器，从而将滤除临时干扰。
3. 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 3.3V 连接执行虚拟读取，从而使干扰为正向，远离误跳闸。如果误跳闸极性反转，则将使用 0V 信号的相反虚拟读取。

公告	在 DCAN FIFO 模式下，所接收到的消息可能以无序状态存放在 FIFO 缓冲器中
受影响版本	0、A、B、C
详细信息	在 DCAN FIFO 模式下，所接收到的、采用相同仲裁和掩码 ID 的消息应按照其被接收的先后顺序存放在 FIFO 中。然后 CPU 通过 IF1/IF2 接口寄存器从 FIFO 检索所接收到的消息。某些消息并不会按照其被接收的先后顺序，存放在 FIFO 中。如果消息的排列顺序对应用程序进行处理至关重要，则此行为将阻止正确使用 DCAN FIFO 模式。
权变措施	无

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。

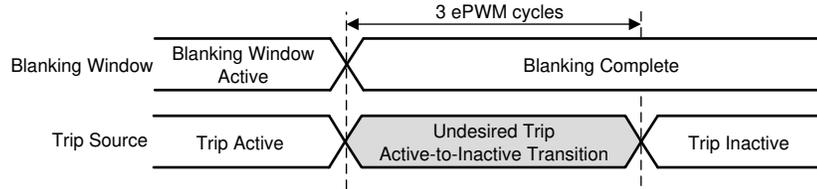


图 3-2. 不良跳闸事件和消隐窗口过期

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。

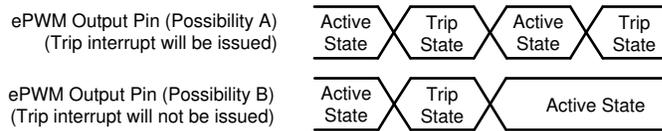


图 3-3. 可能会产生不良的 ePWM 输出

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

ePWM : 在启动消隐窗口后的前 3 个周期内, 消隐窗口不会过滤跳闸事件

受影响版本

0、A、B、C

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 DCxEVType 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。

公告**eQEP: 索引期间方向变化时位置计数器复位错误****受影响版本**

0、A、B、C

详细信息

在使用 $PCRM = 0$ 的配置时，若索引输入处于活动状态时方向发生变化，则位置计数器 (QPOSCNT) 可能会错误地复位，从而导致计数器值发生意外变化。若如此，可能会与位置计数器的预期值相差多达 ± 4 个数值，并随后意外设置错误标志。

在使用 $PCRM = 0$ 的配置时[即发生索引事件时位置计数器复位 (QEPCTL[PCRM] = 00)]，若在正向运动期间发生索引事件，则位置计数器在下一个 eQEP 时钟时复位为 0。若在反向移动期间发生索引事件，则位置计数器将在下一个 eQEP 时钟时复位为 QPOS MAX 寄存器中的值。eQEP 外围设备会记录第一个索引标记 (QEPSTS[FIMF]) 的出现情况以及 QEPSTS 寄存器中第一个索引事件标记 (QEPSTS[FIDF]) 方向。此外，其还会记录第一个索引标记上的正交边沿以便在索引事件复位操作中使用相同的相对正交跃迁。

如果在索引脉冲活动时方向发生改变，则该模块仍将继续查找相对正交跃迁以执行位置计数器复位操作。若如此，位置计数器的值会发生意外变化。

未同时发生方向改变的下一个索引事件将正确复位计数器并按预期工作。

权变措施

在索引处于活动状态时，若方向发生变化且位置计数器值的相应变化可能会影响应用程序，请勿使用 $PCRM = 0$ 的配置。

若适用于此应用程序，则位置计数器复位所运用的其他选项[例如索引事件初始化(IEI)]不会出现此问题。

公告	闪存：不生成单个位 ECC 错误中断
受影响版本	0、A、B、C
详细信息	如果单个位 ECC 错误阈值配置为 0，则当存在单个位错误时不会生成单个位错误中断。
权变措施	将错误阈值位字段 (FLASH_ECC_REGS.ERR_THRESHOLD.ERR_THRESHOLD 字段) 设置为大于或等于 1 的值。请注意：阈值位字段的默认值为 0。

公告	闪存 API：C2000Ware 4.03.00.00 中量产版本不可用
受影响版本	0、A、B、C
详细信息	C2000Ware 4.03.00.00 中发布的闪存 API 版本 2.00.00.00 是测试状态。量产状态闪存 API 版本为 2.00.01.00 (或更高版本)，可用于 C2000Ware 版本 5.00.00.00 (或更高版本) 中。
权变措施	将闪存 API 版本 2.00.01.00 (或更高版本) 用于任何生产应用程序或闪存编程软件。

公告

FPU : FPU 至 CPU 寄存器移动操作之前是任何 FPU 2p 操作

受影响版本

0、A、B、C

详细信息

当多周期 (2p) FPU 指令后跟 FPU 到 CPU 寄存器转移时，此公告适用。如果 FPU 到 CPU 读取指令源寄存器与 2p 指令目标相同，则在 2p 指令完成之前读取的可能是 FPU 寄存器的值。这是因为 2p 指令依赖于流水线 E3 阶段期间结果的数据转发。如果在 E3 阶段碰巧发生流水线停滞，则读取指令不会及时转发结果。

受此公告影响的 2p 指令是 MPYF32、ADDF32、SUBF32 和 MACF32。FPU 寄存器读取的目标必须是一个 CPU 寄存器 (ACC、P、T、XAR0...XAR7)。如果寄存器读取是 FPU 到 FPU 寄存器转移，则此公告不适用。

在下面的示例中，2p 指令 MPYF32 使用 R6H 作为其目标。FPU 寄存器读取 MOV32 使用同一个寄存器 R6H 作为其源，并使用 CPU 寄存器作为目标。如果在 E3 流水线阶段发生停滞，则 MOV32 将在 MPYF32 指令完成之前读取 R6H 的值。

问题示例：

```

MPYF32 R6H, R5H, R0H ; 2p FPU instruction that writes to R6H
|| MOV32 *XAR7++, R4H
F32TOUI16R R3H, R4H ; delay slot
ADDF32 R2H, R2H, R0H
|| MOV32 *--SP, R2H ; alignment cycle
MOV32 @XAR3, R6H ; FPU register read of R6H
    
```

图 3-4 展示了流水线中没有停滞时的问题流水线图。

Instruction	F1	F2	D1	D2	R1	R2	E	W	Comments	
	FPU pipeline-->				R1	R2	E1	E2		E3
I1 MPYF32 R6H, R5H, R0H MOV32 *XAR7++, R4H	I1									
I2 F32TOUI16R R3H, R4H	I2	I1								
I3 ADDF32 R3H, R2H, R0H MOV32 *--SP, R2H	I3	I2	I1							
I4 MOV32 @XAR3, R6H	I4	I3	I2	I1						
		I4	I3	I2	I1					
			I4	I3	I2	I1				
				I4	I3	I2	I1			
					I4	I3	I2	I1		
						I4	I3	I2	I1	I4 samples the result as it enters the R2 phase. The product R6H=R5H*R0H (I1) finishes computing in the E3 phase, but is forwarded as an operand to I4. This makes I4 appear to be a 2p instruction, but I4 actually takes 3p cycles to compute.
							I4	I3	I2	
								I4	I3	

图 3-4. 流水线中没有停滞时的问题流水线图

公告 (续)

FPU : FPU 至 CPU 寄存器移动操作之前是任何 FPU 2p 操作

图 3-5 展示了指令 I1 的 E3 时隙中存在停滞时的问题流水线图。

	Instruction	F1	F2	D1	D2	R1	R2	E	W	E3	Comments
		FPU pipeline-->				R1	R2	E1	E2		
I1	MPYF32 R6H, R5H, R0H MOV32 *XAR7++, R4H	I1									
I2	F32TOUI16R R3H, R4H	I2	I1								
I3	ADDF32 R3H, R2H, R0H MOV32 *--SP, R2H	I3	I2	I1							
I4	MOV32 @XAR3, R6H	I4	I3	I2	I1						
			I4	I3	I2	I1					
				I4	I3	I2	I1				
					I4	I3	I2	I1			
						I4	I3	I2	I1		
							I4	I3	I2	I1 (STALL)	I4 samples the result as it enters the R2 phase, but I1 is stalled in E3 and is unable to forward the product of R5H*R0H to I4 (R6H does not have the product yet due to a design bug). So, I4 reads the old value of R6H.
							I4	I3	I2	I1	There is no change in the pipeline as it was stalled in the previous cycle. I4 had already sampled the old value of R6H in the previous cycle.
								I4	I3	I2	Stall over

图 3-5. 指令 I1 的 E3 时隙中存在停滞时的问题流水线图

权变措施

在本例中，应将 MPYF32、ADDF32、SUBF32 和 MACF32 视为 3p 周期指令。必须将三个 NOP 或非冲突指令放置在指令的延迟时隙中。

C28x 代码生成工具 v.6.2.0 及更高版本都将生成正确的指令序列并检测汇编代码中的错误。在以前的版本 v6.0.5 (适用于 6.0.x 分支) 和 v.6.1.2 (适用于 6.1.x 分支) 中，编译器将生成正确的指令序列，但汇编器不会检测汇编代码中的错误。

解决方法示例：

```

MPYF32 R6H, R5H, R0H
|| MOV32 *XAR7++, R4H      ; 3p FPU instruction that writes to R6H
F32TOUI16R R3H, R4H      ; delay slot
ADDF32 R2H, R2H, R0H
|| MOV32 *--SP, R2H      ; delay slot
NOP                       ; alignment cycle
MOV32 @XAR3, R6H         ; FPU register read of R6H
    
```

图 3-6 展示了带解决方法的流水线图。

公告 (续)

FPU : FPU 至 CPU 寄存器移动操作之前是任何 FPU 2p 操作

	Instruction	F1	F2	D1	D2	R1	R2	E	W	Comments
		FPU pipeline-->				R1	R2	E1	E2	
I1	MPYF32 R6H, R5H, R0H MOV32 *XAR7++, R4H	I1								
I2	F32TOUI16R R3H, R4H	I2	I1							
I3	ADDF32 R3H, R2H, R0H MOV32 *--SP, R2H	I3	I2	I1						
I4	NOP	I4	I3	I2	I1					
I5	MOV32 @XAR3, R6H	I5	I4	I3	I2	I1				
			I5	I4	I3	I2	I1			
				I5	I4	I3	I2	I1		
					I5	I4	I3	I2	I1 (STALL)	Due to one extra NOP, I5 does not reach R2 when I1 enters E3; thus, forwarding is not needed.
					I5	I4	I3	I2	I1	There is no change due to the stall in the previous cycle.
						I5	I4	I3	I2	I1 moves out of E3 and I5 moves to R2. R6H has the result of R5H*R0H and is read by I5. There is no need to forward the result in this case.
							I5	I4	I3	

图 3-6. 带解决方法的流水线图

公告

GPIO19/X1 : 若未按照推荐的运行条件而施加电压, 可能会导致器件出错

受影响版本

0、A、B、C

详细信息

如果对 GPIO19/X1 引脚施加的电压超过推荐的运行条件 (高于 VDDIO 或低于 VSS) , 则器件可能无法正常运行。并造成下列影响:

- 在上电和/或 XRSn 置为无效后, 器件不能退出引导 ROM
- 器件无法通过 JTAG 连接
- 应用程序代码执行期间对器件时钟频率产生影响

权变措施

请特别注意该器件上 GPIO19/X1 引脚的布局和布线。虽然该引脚上的电压超出规格要求是导致出现该问题的原因; 但电压过高也可能是由连接到引脚的外部元件或 PCB 上其他源的噪声耦合引起的。如果将单端时钟应用于 GPIO19, 则外部时钟驱动器的阻抗与此引脚的特性不匹配也可能导致过压。建议进行信号完整性验证, 以确定施加的电压在数据表容差范围内。

如果未在系统中使用此引脚, 则数据表中关于未使用引脚的建议仍然适用:

- 无连接 (启用内部上拉的输入模式)
- 无连接 (禁用内部上拉的输出模式)
- 上拉或下拉电阻器 (任意值电阻器, 输入模式, 禁用内部上拉)

然而, 如果 PCB 设计允许, 则最好修改最后一个要点, 以最大限度地减少噪声对该引脚的影响:

- 将该引脚直接连接到 VSS。

公告
IC2 : 目标发送器模式、标准模式 SDA 时序限制
受影响版本

0、A、B、C

详细信息

MCU 上的 I2C 外设是一款快速模式器件；当与标准模式主机一同使用时，其将对 SCL（时钟）线路执行时钟延展。

I2C 规范对于标准模式系统中使用的快速模式器件有一项要求，即在释放 SCL 线路之前满足 $t_{\text{SU:DAT}}$ （数据设置时间）+ $t_{\text{r(max)}}$ （上升时间）。请参阅 NXP 半导体 *I²C 总线规范和用户手册(UM10204)* 中“标准、快速和快速模式 Plus I²C 总线器件的 SDA 和 SCL 总线特性”表的脚注 4。

然而，在上述情况下，C2000 I2C 时钟会将 SCL 线延展到一个固定量 = $6 * f_{\text{mod}}$ 时钟（C2000 的 I2C 时钟速率）。当 C2000™ 微控制器充当标准模式主机的目标发送器时，如果 SDA 的 t_{r} 过长，则 C2000 可能会在数据 (SDA) 就绪之前释放时钟线路 (SCL)。

NXP Semiconductors *I²C 总线规范和用户手册(UM10204)* 中的“上拉电阻器大小调整”部分提供了有关根据上升时间 (t_{r}) 和总线电容 (C_{b}) 选择适当 pu 电阻器 (R_{p}) 的更多详细信息，如 [方程式 1](#) 所示。

$$R_{\text{p(max)}} = \frac{t_{\text{r}}}{0.8473 \times C_{\text{b}}} \quad (1)$$

权变措施
1. 通过使用强上拉

减少 t_{r} 来确保满足 $t_{\text{SU:DAT}} + t_{\text{r(max)}}$ ，用户可以在 SDA 线路上配置上拉电阻，以便根据其系统中的 f_{mod} 时钟值，满足的“SDA 数据上升时间要求”列中列出的限制 [表 3-3](#)。这将确保在 C2000 释放 SCL 信号时，SDA 线路上存在的数据有效。

[表 3-4](#) 给出了在给定 f_{mod} 时钟 (MHz) 和 C_{b} （总线电容）下的建议 R_{p} 电阻值。对于其他 C_{b} 值，请使用 [方程式 1](#) 来计算系统中所需的 R_{p} 值。

表 3-3. C2000 作为带有标准模式主机的目标发送器时的数据上升时间要求

f_{mod} 时钟 (MHz)	f_{mod} 周期 (ns)	C2000 I2C 的 SCL 时钟延展延迟 (ns) : ($6 * f_{\text{mod}}$ 时钟)	数据建立时间 (ns) : $t_{\text{SU:DAT}}$ (标准模式)	SDA 数据上升时间要求 (ns) : t_{r}
7	142.9	857	250	607
8	125	750		500
9	111	666		416
10	100	600		350
11	90.9	545		295
12	83.3	500		250

公告 (续)

IC2 : 目标发送器模式、标准模式 SDA 时序限制

表 3-4. 公共总线电容 (C_b) 的上拉电阻 (R_p) 值

f_{mod} 时钟 (MHz)	SDA 数据上升时间要求 (ns) : t_r	R_p (k Ω), 对于 $C_b = 100$ pF	R_p (k Ω), 对于 $C_b = 200$ pF	R_p (k Ω), 对于 $C_b = 300$ pF	R_p (k Ω), 对于 $C_b = 400$ pF
7	607	7.1	3.5	2.3	1.7
8	500	5.9	2.9	1.9	1.4
9	416	4.9	2.4	1.6	1.2
10	350	4.1	2.0	1.3	1.0
11	295	3.4	1.7	1.1	0.8
12	250	2.9	1.4	0.9	0.7

2. $T_r = 1000\text{ns}$

由于一般 I2C 使用方面的限制、因此不推荐使用此权变措施、请尽可能使用权变措施 1。

如果系统在 SDA 线路上需要 1000 ns 的上升时间、则可以将 C2000 I2C f_{mod} 时钟配置为 4.8 MHz, 以便时钟延展 ($6 * f_{\text{mod}}$ 时钟) 满足该要求。由此可得, $t_r = (1/4.8 \text{ MHz}) * 6 = 1000 \text{ ns}$ 。此权变措施仅在 C2000 I2C 是 I2C 总线上目标的有效系统中有效。请注意, 4.8 MHz 超出了数据表中 f_{mod} 时钟 7 MHz 至 12 MHz 的所需范围。在 4.8 MHz 时使用 f_{mod} , 尽管其已超出数据表的所需范围, 也将适用于标准模式主机总线上的目标模式下的 C2000 I2C。除了此权变措施中列出的配置外, 在任何其他配置中使用 $f_{\text{mod}} = 4.8 \text{ MHz}$ 都将导致违反其他时序参数的要求, 因此不允许如此操作。

公告 **存储器：在有效存储器之外进行预取**

受影响版本 0、A、B、C

详细信息 C28x CPU 预取指令的范围超出其流水线中当前正在活动的指令范围。如果预取发生在有效存储器结束之后，则 CPU 可能会接收到无效的操作码。

权变措施 **M1** - 预取队列的深度为 8 x16 字。因此，代码不应在有效存储器末尾的 8 个字以内。可以在两个有效存储器块之间实现跨边界预取。

示例 1：M1 在地址 0x7FF 处结束，后面不跟随另一个存储器块。M1 中代码的存储地址不应超过 0x7F7。地址 0x7F8-0x7FF 不应用于代码。

示例 2：M0 结束于地址 0x3FF，有效存储器 (M1) 紧随其后。M0 中的代码可存储在 0x3FF 及以下的地址。代码也可以交叉进入 M1，最高到地址 0x7F7 (含地址 0x7F7)。

表 3-5. 受公告影响的存储器

内存类型	受影响的地址
M1	0x0000 07F8-0x0000 07FF

公告

MPOST : 在某些早期材料上不会执行存储器开机自检

受影响版本

0、A、B、C

详细信息

MPOST (存储器开机自检) 可用于功能安全应用场景，以便在开机时测试器件存储器。通过使用 DCSM 安全工具写入 Z1_GPREG2.MPOST 位来激活此功能。在受影响的材料上，即使写入了 Z1_GPREG2.MPOST 位，MPOST 也不会执行。

权变措施

- **检查 OTP 修订版本**：固定材料的 OTP 修订版本号将大于 2。OTP 版本号可以使用表 3-6 来确定。MPOST 的工作原理如 [TMS320F280013x 实时微控制器技术参考手册](#) 中所述。
- **F280013x SDL 中的等效存储器测试**：使用 F280013x 软件诊断库 (SDL) 中包含的 *STA_March* 函数，可以对可从主应用程序执行的存储器进行等效测试。SDL 包含在 C2000Ware 安装程序中，位于以下父目录中：*C:/ti/c2000/C2000Ware_5_02_00_00/libraries/diagnostic/f280013x/*。查看“examples”文件夹中的“test application”项目，以及有关如何调用此存储器检查的“docs”子文件夹中 STL 的说明。

表 3-6. OTP 修订版本号位置

地址	8 位 MSB	8 位 LSB
0x0007 11DE	0x5A	OTP 版本

公告**系统：多次连续写入 CLKSRCCTL1 可能会导致系统挂起****受影响版本**

0、A、B、C

详细信息

在写入 CLKSRCCTL1 寄存器时，在写入操作之间多次未延迟，则系统可能会发生挂起的情况，并且只能通过外部 XRSn 复位或 Watchdog 复位才能恢复。在 SYSCLK 与 OSCCLKSRCSEL 所选时钟之间的时钟比有问题时，就会出现这种情况，然而，并非每次都会出现这个问题。

如果在使用调试器时遇到此问题，那么在点击暂停后，程序计数器将位于 Boot ROM 复位向量处。

实施此权变措施，可避免由 SYSCLK 与 OSCCLK 时钟比引起的这种情况。

权变措施

每次写入 CLKSRCCTL1 寄存器后，使用 NOP 指令添加 300 个 SYSCLK 周期的软件延迟。

示例：

```

ClkCfgRegs.CLKSRCTL1.bit.INTOSC2OFF=0;           // Turn on INTOSC2
asm(" RPT #250 || NOP");                          // Delay of 250 SYSCLK cycles
asm(" RPT #50 || NOP");                            // Delay of 50 SYSCLK cycles
ClkCfgRegs.CLKSRCTL1.bit.OSCCLKSRCSEL = 0;        // Clk Src = INTOSC2
asm(" RPT #250 || NOP");                          // Delay of 250 SYSCLK cycles
asm(" RPT #50 || NOP");                            // Delay of 50 SYSCLK cycles

```

C2000Ware_3_00_00_00 和更高版本将实施此权变措施。

公告	PLL 基准时钟丢失检测：时钟丢失标志可能错误激活
受影响版本	0、A、B、C
详细信息	寄存器位 <code>SYSPLLSTS.REF_LOST</code> 可能设置错误，因此错误地指示基准时钟丢失。
权变措施	<p>请勿使用 PLL 基准时钟丢失检测功能。或者，可以使用时钟丢失检测(MCD)功能或双时钟比较器(DCC)来检测基准时钟丢失。</p> <ul style="list-style-type: none">• 对于 MCD 方法、请参阅 <code>SYSCTL</code> 文件夹下的 C2000Ware 示例 <code>sysctl_ex1_missing_clock_detection</code>。• 对于 DCC 方法、请参阅 <code>DCC</code> 文件夹下的 C2000Ware 示例 <code>dcc_ex4_clock_fail_detect</code>。

公告	看门狗：WDKEY 寄存器不受 EALLOW 保护
受影响版本	0、A、B、C
详细信息	WDKEY 寄存器不受 EALLOW 保护。无需发出 EALLOW 和 EDIS 指令来写入该寄存器。要在 WDKEY 受 EALLOW 保护的其他器件上启用软件重用，建议使用 EALLOW 和 EDIS。
权变措施	无

4 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。

4.1 器件修订版本 B 使用说明

在更高版本的器件上找到了器件版本适用的使用说明。有关更多详细信息，请参阅[器件版本 C 使用说明](#)。

4.2 器件修订版本 B 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 C 公告](#)》。

5 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。

5.1 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[器修订版本 C 的使用说明](#)。

5.2 器件修订版本 A 公告

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

6 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。

6.1 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 C 的使用说明](#)。

6.2 器件修订版本 0 公告

在后续的器件修订版本中，可参阅器件修订版本适用的公告。欲了解更多详细信息、请参阅[器件修订版本 C 公告](#)。

7 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 TMS320F280013x 器件的更多信息，请参阅以下文档：

- [TMS320F280013x 实时微控制器数据表](#)
- [TMS320F280013x 实时微控制器技术参考手册](#)

8 商标

C2000™ is a trademark of Texas Instruments.

所有商标均为其各自所有者的财产。

9 修订历史记录

Changes from MAY 18, 2024 to DECEMBER 9, 2024	Page
• 此“修订历史记录”列出了从 SPRZ506C 到 SPRZ506D 的更改。.....	3
• 新增了 GPIO19/X1 ：若未按照推荐的运行条件而施加电压，可能会导致器件出错 公告。.....	21
• 添加了 MPOST ：在某些早期材料上不会执行存储器开机自检的公告。.....	25

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