

Programmer's Guide

DS160PR810 编程指南



摘要

本文档是 [DS160PR810](#) 八通道 PCI-Express 第 4 代线性转接驱动器的编程参考。本文档包含与 [DS160PR810](#) 高级配置选项相关的详细信息。目标受众包括从事系统诊断和控制软件工作的软件工程师。

TI 建议读者熟悉 [DS160PR810](#) 数据表。本文档以及与 [DS160PR810](#) 转接驱动器相关的所有其他配套资料数据 (应用手册、模型等) 可从 TI 网站下载。此外, 可以联系您当地的德州仪器 (TI) 现场销售代表。

内容

1 访问方法	2
1.1 通过 SMBus 对寄存器进行编程.....	2
1.2 通过外部 EEPROM 进行器件配置.....	2
2 寄存器映射概述	3
3 编程序列示例	4
3.1 设置 CTLE 增益电平.....	4
3.2 复位 RX 检测状态机.....	6
3.3 设置 CTLE 直流增益电平.....	6
3.4 设置 VOD 电平.....	7
4 共享寄存器	8
5 通道寄存器	10
6 参考资料	12
7 修订历史记录	12

1 访问方法

设计人员可以通过两种方式来访问 DS160PR810。这两种方法为：

- 通过串行管理总线 (SMBus) 进行寄存器控制
- 通过外部 EEPROM 进行自动配置

1.1 通过 SMBus 对寄存器进行编程

可通过标准 SMBus 协议访问 DS160PR810 内部寄存器。DS160PR810 具有两个通道组，即组 0 (通道 0 - 3) 和组 1 (通道 4-7)，每个通道都具有单独的寄存器集并需要唯一的 SMBus 从地址。在上电时会根据 EQ0_0 / ADDR1 和 EQ1_0 / ADDR0 引脚的配置来确定 SMBus 从地址对 (每个通道组一个)。当内部上电复位信号置为无效后，上电时读取引脚状态。

EQ0_0 / ADDR1 和 EQ1_0 / ADDR0 引脚以及 MODE、GAIN0、GAIN1、EQ0_1、EQ1_1 和 RX_DET 引脚均为 4 级输入引脚，用于控制器件的配置。这些四电平输入使用电阻分压器来帮助设置四个有效电平，如表 1-1 所示。

表 1-1. DS160PR810 4 级控制引脚设置

引脚电平	引脚设置
L0	1kΩ 至 GND
L1	13kΩ 至 GND
L2	59kΩ 至 GND
L3	浮点

可通过在 EQ0_0 / ADDR1 和 EQ1_0 / ADDR0 引脚上放置外部电阻器搭接，可以为器件分配 16 个唯一的 SMBus 从地址对 (每个通道组一个地址)，如表 1-2 所示。当多个 DS160PR810 器件位于同一 SMBus 接口总线上时，每个器件的每个通道组必须配置一个唯一的 SMBus 从地址对。

表 1-2. DS160PR810SMBus 地址映射

ADDR1 引脚电平	ADDR0 引脚电平	组 0 : 通道 0-3 : 7 位地址 [HEX]	组 1 : 通道 4-7 : 7 位地址 [HEX]
L0	L0	0x18	0x19
L0	L1	0x1A	0x1B
L0	L2	0x1C	0x1D
L0	L3	0x1E	0x1F
L1	L0	0x20	0x21
L1	L1	0x22	0x23
L1	L2	0x24	0x25
L1	L3	0x26	0x27
L2	L0	0x28	0x29
L2	L1	0x2A	0x2B
L2	L2	0x2C	0x2D
L2	L3	0x2E	0x2F
L3	L0	0x30	0x31
L3	L1	0x32	0x33
L3	L2	0x34	0x35
L3	L3	0x36	0x37

1.2 通过外部 EEPROM 进行器件配置

DS160PR810 可在上电时自动从 EEPROM 读取其初始配置。有关此器件的 EEPROM 十六进制文件生成的详细信息，请参阅 [了解 DS160PR810 PCI-Express 4.0 线性转接驱动器的 EEPROM 编程应用报告](#)。

2 寄存器映射概述

DS160PR810 具有两种类型的寄存器：

- **共享寄存器** - 这种寄存器可随时访问，用于器件级配置、状态回读、控制或读回器件 ID 信息。
- **通道寄存器** - 这种寄存器用于控制和配置每个单独通道的特定功能。所有通道采用同一通道寄存器组，可单独进行配置。

单个通道组的共享寄存器和通道寄存器都包含在单个寄存器页面中，如表 2-1 所示。

表 2-1. 寄存器映射概述

地址范围	通道组 0 访问	通道组 1 访问
0x00 - 0x03	通道 0 寄存器	通道 4 寄存器
0x20 - 0x23	通道 1 寄存器	通道 5 寄存器
0x40 - 0x43	通道 2 寄存器	通道 6 寄存器
0x60 - 0x63	通道 3 寄存器	通道 7 寄存器
0x80 - 0x83	广播写入通道组 0 寄存器、读取通道 0 寄存器	广播写入通道组 1 寄存器、读取通道 4 寄存器
0xA0 - 0xA3	广播写入通道 0-1 寄存器、读取通道 0 寄存器	广播写入通道 4-5 寄存器、读取通道 4 寄存器
0xC0 - 0xC3	广播写入通道 2-3 寄存器、读取通道 2 寄存器	广播写入通道 6-7 寄存器、读取通道 6 寄存器
0xE2 - 0xE3、 0xF0 - 0xF1	组 0 共享寄存器	组 1 共享寄存器

3 编程序列示例

DS160PR810 具有高度的可编程性，并可针对多种应用进行定制。以下各节提供了针对某些常见应用对 DS160PR810 进行编程的指导。

每个序列中提供了以下信息：

- **步骤**：很多序列包含多个步骤。执行操作的顺序由步骤编号指示。
- **寄存器组**：操作旨在用于组 0 或组 1 的共用寄存器或通道寄存器。由于每个通道组都有自己的 SMBus 地址，因此无需明确选择通道组。
- **操作**：读取或写入。如果是读取操作，则与寄存器值或写入掩码无关。
- **寄存器地址**：选择要写入的寄存器。
- **寄存器值**：要写入寄存器地址的值。
- **写入掩码**：除非写入掩码为 0xFF，否则将所有写操作都作为读取/修改/写入操作执行。仅修改由掩码标识的位。

3.1 设置 CTLE 增益电平

DS160PR810 需要手动 CTLE 调优。可通过修改每个 CTLE 级 (EQ1 和 EQ2) 的值或绕过 EQ1 级来更改 CTLE 增益电平。CTLE 电平可针对每个通道单独设定或者广播至所有通道。表 3-1 显示了一个示例序列，通过对每个通道单独执行写入操作，将组 0 通道上的 CTLE 增益电平设置为 5.0dB (8GHz 时) (CTLE 指数 2)，并将组 1 通道上的 CTLE 增益电平设置为 7.0dB (CTLE 指数 4)。使用表 3-3 中提供的寄存器值将 CTLE 增益电平设置为任何其他可用值。

表 3-1. 用于分别在每个通道上设置 CTLE 电平的序列

步骤	寄存器组	操作	寄存器地址 [十六进制]	寄存器值 [十六进制]	写入掩码 [十六进制]	备注
1	组 0：通道 0	写入	0x01	0x08	0x3F	将通道 0 上的 CTLE 设为指数 2。
2	组 0：通道 1	写入	0x21	0x08	0x3F	将通道 1 上的 CTLE 设为指数 2。
3	组 0：通道 2	写入	0x41	0x08	0x3F	将通道 2 上的 CTLE 设为指数 2。
4	组 0：通道 3	写入	0x61	0x08	0x3F	将通道 3 上的 CTLE 设为指数 2。
5	组 1：通道 4	写入	0x01	0x11	0x3F	将通道 4 上的 CTLE 设为指数 4。
6	组 1：通道 5	写入	0x21	0x11	0x3F	将通道 5 上的 CTLE 设为指数 4。
7	组 1：通道 6	写入	0x41	0x11	0x3F	将通道 6 上的 CTLE 设为指数 4。
8	组 1：通道 7	写入	0x61	0x11	0x3F	将通道 7 上的 CTLE 设为指数 4。

假设 0x18 和 0x19 分别是通道组 0 和 1 的 SMBus 地址，以下是表 3-1 中序列的 XML 批处理脚本：

```
<i2c_write addr="0x18" count="0" radix"16">01 08</i2c_write>
<i2c_write addr="0x18" count="0" radix"16">21 08</i2c_write>
<i2c_write addr="0x18" count="0" radix"16">41 08</i2c_write>
<i2c_write addr="0x18" count="0" radix"16">61 08</i2c_write>
<i2c_write addr="0x19" count="0" radix"16">01 11</i2c_write>
<i2c_write addr="0x19" count="0" radix"16">21 11</i2c_write>
<i2c_write addr="0x19" count="0" radix"16">41 11</i2c_write>
<i2c_write addr="0x19" count="0" radix"16">61 11</i2c_write>
```

表 3-2 显示了一个示例序列，通过对每个通道组单独执行广播写入操作，将组 0 通道上的 CTLE 增益电平设置为 5.0dB (8GHz 时) (CTLE 指数 2)，并将组 1 通道上的 CTLE 增益电平设置为 7.0dB (CTLE 指数 4)。

表 3-2. 用于向所有通道广播 CTLE 电平的序列

步骤	寄存器组	操作	寄存器地址 [十六进制]	寄存器值 [十六进制]	写入掩码 [十六进制]	备注
1	组 0 : 通道 0-3	写入	0x81	0x08	0x3F	将通道 0-3 上的 EQ 设置为指数 2。
2	组 1 : 通道 4-7	写入	0x81	0x11	0x3F	将通道 4-7 上的 EQ 设置为指数 4。

假设 0x18 和 0x19 分别是通道组 0 和 1 的 SMBus 地址，以下是表 3-2 中序列的 XML 批处理脚本：

```
<i2c_write addr="0x18" count="0" radix"16">81 08</i2c_write>
<i2c_write addr="0x19" count="0" radix"16">81 11</i2c_write>
```

表 3-3 提供了 CTLE 控制寄存器值与 CTLE 指数 (0 - 15) 之间的函数关系图。表 3-1 和表 3-2 中提供了示例 CTLE 控制寄存器地址。

表 3-3. CTLE 控制寄存器值与 CTLE 指数之间的函数关系图

CTLE 指数	4GHz 时的 CTLE 增益 (dB)	8GHz 时的 CTLE 增益 (dB)	CTLE 控制寄存器值 [十六进制]
0	0.0	-0.2	0x40
1	1.5	4.5	0x43
2	2.0	5.5	0x08
3	2.5	6.5	0x0A
4	2.7	7.0	0x11
5	3.0	8.0	0x12
6	4.0	9.0	0x13
7	5.0	10.0	0x1A
8	6.0	11.0	0x1B
9	7.0	12.0	0x23
10	7.5	13.0	0x2B
11	8.0	13.5	0x2C
12	8.5	15.0	0x2D
13	9.5	16.5	0x35
14	10.0	17.0	0x36
15	11.0	18.0	0x3F

3.2 复位 RX 检测状态机

可为所有通道手动复位 RX 检测状态机。

表 3-4. 用于复位 RX 检测状态机的序列

步骤	寄存器组	操作	寄存器地址 [十六进制]	寄存器值 [十六进制]	写入掩码 [十六进制]	备注
1	组 0 : 通道 0-3	写入	0x89	0x04	0x04	将 RX 检测复位位设置为 : 1 (复位 RX 检测状态机)
2	组 1 : 通道 4-7	写入	0x89	0x00	0x04	将 RX 检测复位位设置为 : 0 (清除设置位)

假设 0x18 和 0x19 分别是通道组 0 和 1 的 I2C/SMBus 地址，以下是表 3-4 中用于复位所有通道上 RX 检测状态机的序列的 XML 批处理脚本：

```
<i2c_write addr="0x18" count="0" radix="16">89 04</i2c_write>
<i2c_write addr="0x19" count="0" radix="16">89 04</i2c_write>
<i2c_write addr="0x18" count="0" radix="16">09 00</i2c_write>
<i2c_write addr="0x19" count="0" radix="16">09 00</i2c_write>
```

3.3 设置 CTLE 直流增益电平

CTLE 直流增益值可针对每个通道单独设定或者广播至所有通道。

表 3-5. 用于向所有通道广播直流增益电平的序列

步骤	寄存器组	操作	寄存器地址 [十六进制]	寄存器值 [十六进制]	写入掩码 [十六进制]	备注
1	组 0 : 通道 0-3	写入	0x82	0x00	0x04	将组 0 通道上的直流增益设置为 : 0dB (默认值)。
				0x04	0x04	将组 0 通道上的直流增益设置为 : 3.5dB。
2	组 1 : 通道 4-7	写入	0x82	0x00	0x04	将组 1 通道上的直流增益设置为 : 0dB (默认值)。
				0x04	0x04	将组 1 通道上的直流增益设置为 : 3.5dB。

假设 0x18 和 0x19 分别是通道组 0 和 1 的 I2C/SMBus 地址，以下是表 3-5 中用于将直流增益广播为 3.5dB 的序列的 XML 批处理脚本：

```
<i2c_write addr="0x18" count="0" radix="16">82 04</i2c_write>
<i2c_write addr="0x19" count="0" radix="16">82 04</i2c_write>
```

3.4 设置 VOD 电平

如果需要，可修改 DS160PR810 驱动器的差分输出电压。

表 3-6. 用于向所有通道广播 VOD 电平的序列

步骤	寄存器组	操作	寄存器地址 [十六进制]	寄存器值 [十六进制]	写入掩码 [十六进制]	备注
1	组 0 : 通道 0-3	写入	0x82	0x00	0x03	将组 0 通道上的 VOD 设置为 : -6dB
				0x01	0x03	将组 0 通道上的 VOD 设置为 : -3.5dB
				0x10	0x03	将组 0 通道上的 VOD 设置为 : -1.6dB
				0x11	0x03	将组 0 通道上的 VOD 设置为 : 0dB (默认值)
2	组 1 : 通道 4-7	写入	0x82	0x00	0x03	将组 1 通道上的 VOD 设置为 : -6dB
				0x01	0x03	将组 1 通道上的 VOD 设置为 : -3.5dB
				0x10	0x03	将组 1 通道上的 VOD 设置为 : -1.6dB
				0x11	0x03	将组 1 通道上的 VOD 设置为 : 0dB (默认值)

假设 0x18 和 0x19 分别是通道组 0 和 1 的 I2C/SMBus 地址，以下是表 3-6 中用于通过广播将 VOD 设为 -3.5dB 的序列的 XML 批处理脚本：

```

<i2c_write addr="0x18" count="0" radix"16">82 01</i2c_write>
<i2c_write addr="0x19" count="0" radix"16">82 01</i2c_write>
  
```

4 共享寄存器

表 4-1 列出了共享寄存器。表 4-1 中未列出的所有寄存器偏移地址都应视为保留的存储单元，并且不应修改寄存器内容。

表 4-1. 共享寄存器

偏移	缩写	寄存器名称	部分
0xE2	通用	通用控制寄存器	查找
0xE3	EE_Status	EEPROM 状态寄存器	查找
0xF0	DEVICE_ID0	器件 ID0 寄存器	查找
0xF1	DEVICE_ID1	器件 ID1 寄存器	转到

复杂的位访问类型经过编码可适应小型表单元。表 4-2 显示了适用于此部分中访问类型的代码。

表 4-2. 访问类型代码

访问类型	代码	说明
读取类型		
R	R	只读访问
写入类型		
R/W/SC	R/W/SC	读/写访问，自清除
复位或默认值		
-n		复位后的值或默认值

4.1 通用寄存器 (偏移 = 0xE2) [复位 = 0x0]

表 4-3 显示了通用寄存器。

返回到[汇总表](#)。

表 4-3. 通用寄存器字段说明

位	字段	类型	复位	说明
7	RESERVED	R	0x0	保留
6	rst_i2c_regs	R/W/SC	0x0	器件复位控制：将所有 I ² C 寄存器复位为默认值（自行清除）。
5	RESERVED	R	0x0	保留
4	RESERVED	R	0x0	保留
3	RESERVED	R	0x0	保留
2	保留	R	0x0	保留
1	RESERVED	R	0x0	保留
0	frc_eepm_rd	R/W/SC	0x0	覆盖 MODE 和 READ_EN_N 状态以强制手动加载 EEPROM 配置。

4.2 EE_Status 寄存器 (偏移 = 0xE3) [复位 = 0x0]

表 4-4 中显示了 EE_Status。

返回到[汇总表](#)。

表 4-4. EE_Status 寄存器字段说明

位	字段	类型	复位	说明
7	eecfg_cmplt	R	0x0	EEPROM 加载完成。
6	eecfg_fail	R	0x0	EEPROM 加载失败。
5	eecfg_atmpt_1	R	0x0	尝试加载 EEPROM 映像的次数。
4	eecfg_atmpt_0	R	0x0	参见 MSB
3	RESERVED	R	0x0	保留
2	保留	R	0x0	保留
1	RESERVED	R	0x0	保留
0	RESERVED	R	0x0	保留

4.3 DEVICE_ID0 寄存器 (偏移 = 0xF0) [复位 = X]

DEVICE_ID0 如表 4-5 所示。

返回到[汇总表](#)。

表 4-5. DEVICE_ID0 寄存器字段说明

位	字段	类型	复位	说明
7	RESERVED	R	0x0	保留
6	RESERVED	R	0x0	保留
5	RESERVED	R	0x0	保留
4	RESERVED	R	0x0	保留
3	device_id0_3	R	X	器件 ID0 [3:1]: 011
2	device_id0_2	R	X	参见 MSB
1	device_id0_1	R	X	参见 MSB
0	RESERVED	R	X	保留

4.4 DEVICE_ID1 寄存器 (偏移 = 0xF1) [复位 = 0x27]

DEVICE_ID1 如表 4-6 所示。

返回到[汇总表](#)。

表 4-6. DEVICE_ID1 寄存器字段说明

位	字段	类型	复位	说明
7	device_id1_7	R	0x0	器件 ID1 : 0010 0111
6	device_id1_6	R	0x0	请参阅 MSB
5	device_id1_5	R	0x1	请参阅 MSB
4	device_id1_4	R	0x0	请参阅 MSB
3	device_id1_3	R	0x0	请参阅 MSB
2	device_id1_2	R	0x1	请参阅 MSB
1	device_id1_1	R	0x1	请参阅 MSB
0	device_id1_0	R	0x1	请参阅 MSB

5 通道寄存器

表 5-1 列出了通道寄存器。表 5-1 中未列出的所有寄存器偏移地址都应视为保留的存储单元，并且不应修改寄存器内容。

请注意，为通道 0 或通道 4 寄存器（通道组 0 或通道组 1）提供了寄存器偏移量。对于通道组 0 上的通道 1、2 和 3 寄存器（或通道组 1 上的通道 5、6 和 7），分别将 0x20、0x40 或 0x60 添加到所提供的偏移量。

表 5-1. 通道寄存器

偏移	缩写	寄存器名称	部分
0x0	RX_DET_STS	接收器检测状态寄存器	查找
0x1	EQ_CTRL	均衡器控制寄存器	查找
0x2	GAIN_CTRL	直流增益和 VOD 控制寄存器	查找
0x3	RX_DET_CTRL1	接收器检测控制寄存器 1	查找
0x9	RX_DET_CTRL2	接收器检测控制寄存器 2	转到

复杂的位访问类型经过编码可适应小型表单元。表 5-2 显示了适用于此部分中访问类型的代码。

表 5-2. 访问类型代码

访问类型	代码	说明
读取类型		
R	R	只读访问。
写入类型		
R/W	R/W	读取/写入访问。
复位或默认值		
-n		复位后的值或默认值。

5.1 RX_DET_STS 寄存器 (偏移 = 0x0) [复位 = 0x0]

表 5-3 中显示了 RX_DET_STS。

返回到[汇总表](#)。

表 5-3. RX_DET_STS 寄存器字段说明

位	字段	类型	复位	说明
7	rx_det_comp_p	R	0x0	Rx 检测正极性状态：0：未检测到 1：检测到该值被锁存。
6	rx_det_comp_n	R	0x0	Rx 检测负极性状态：0：未检测到 1：检测到该值被锁存。
5	RESERVED	R	0x0	保留
4	RESERVED	R	0x0	保留
3	RESERVED	R	0x0	保留
2	保留	R	0x0	保留
1	RESERVED	R	0x0	保留
0	RESERVED	R	0x0	保留

5.2 EQ_CTRL 寄存器 (偏移 = 0x1) [复位 = 0x9]

表 5-4 中显示了 EQ_CTRL。

返回到[汇总表](#)。

表 5-4. EQ_CTRL 寄存器字段说明

位	字段	类型	复位	说明
7	RESERVED	R	0x0	保留
6	eq_en_bypass	读/写	0x0	启用 CTLE 1 级旁路： 0：禁用旁路 1：旁路启用
5	eq_bst1_2	读/写	0x0	CTLE 升压 1 级控制。
4	eq_bst1_1	读/写	0x0	请参阅 MSB
3	eq_bst1_0	读/写	0x1	请参阅 MSB
2	eq_bst2_2	读/写	0x0	CTLE 升压 2 级控制。
1	eq_bst2_1	读/写	0x0	请参阅 MSB
0	eq_bst2_0	读/写	0x1	请参阅 MSB

5.3 GAIN_CTRL 寄存器 (偏移 = 0x2) [复位 = 0x3]

表 5-5 中显示了 GAIN_CTRL。

返回到[汇总表](#)。

表 5-5. GAIN_CTRL 寄存器字段说明

位	字段	类型	复位	说明
7	RESERVED	R	0x0	保留
6	RESERVED	R	0x0	保留
5	RESERVED	R	0x0	保留
4	RESERVED	R	0x0	保留
3	RESERVED	R	0x0	保留
2	eq_hi_gain	读/写	0x0	设置 CTLE 直流增益： 0：0dB (推荐) 1：3.5dB
1	drv_sel_vod_1	读/写	0x1	TX VOD 选择： 00：-6dB 01：-3.5dB 10：-1.6dB 11：0dB (推荐)
0	drv_sel_vod_0	读/写	0x1	请参阅 MSB

5.4 RX_DET_CTRL1 寄存器 (偏移 = 0x3) [复位 = 0x0]

表 5-6 中显示了 RX_DET_CTRL1。

返回到[汇总表](#)。

表 5-6. RX_DET_CTRL1 寄存器字段说明

位	字段	类型	复位	说明
7	RESERVED	R	0x0	保留
6	RESERVED	R	0x0	保留
5	RESERVED	R	0x0	保留
4	RESERVED	R	0x0	保留
3	RESERVED	R	0x0	保留
2	mr_rx_det_man	读/写	0x0	手动覆盖 rx_detect。
1	en_rx_det_count	读/写	0x0	启用 RX 检测有效计数器。
0	sel_rx_det_count	读/写	0x0	启用前选择有效的检测计数： 0：2 次连续有效检测 1：3 次连续有效检测

5.5 RX_DET_CTRL2 寄存器 (偏移 = 0x9) [复位 = 0x0]

表 5-7 中显示了 RX_DET_CTRL2。

返回到[汇总表](#)。

表 5-7. RX_DET_CTRL2 寄存器字段说明

位	字段	类型	复位	说明
7	RESERVED	R	0x0	保留
6	RESERVED	R	0x0	保留
5	RESERVED	R	0x0	保留
4	RESERVED	R	0x0	保留
3	RESERVED	R	0x0	保留
2	mr_rx_det_rst	读/写	0x0	RX 检测复位
1	RESERVED	R	0x0	保留
0	RESERVED	R	0x0	保留

6 参考资料

1. [DS160PR810 八通道 PCI-Express 4.0 线性转接驱动器 数据表](#)
2. [了解 DS160PR810 PCI-Express 4.0 线性转接驱动器的 EEPROM 编程 应用报告](#)

7 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

日期	修订版本	说明
November 2023	*	初始发行版

重要声明和免责声明

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 产品的应用。严禁对这些资源进行其他复制或展示。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

Copyright © 2023，德州仪器 (TI) 公司