

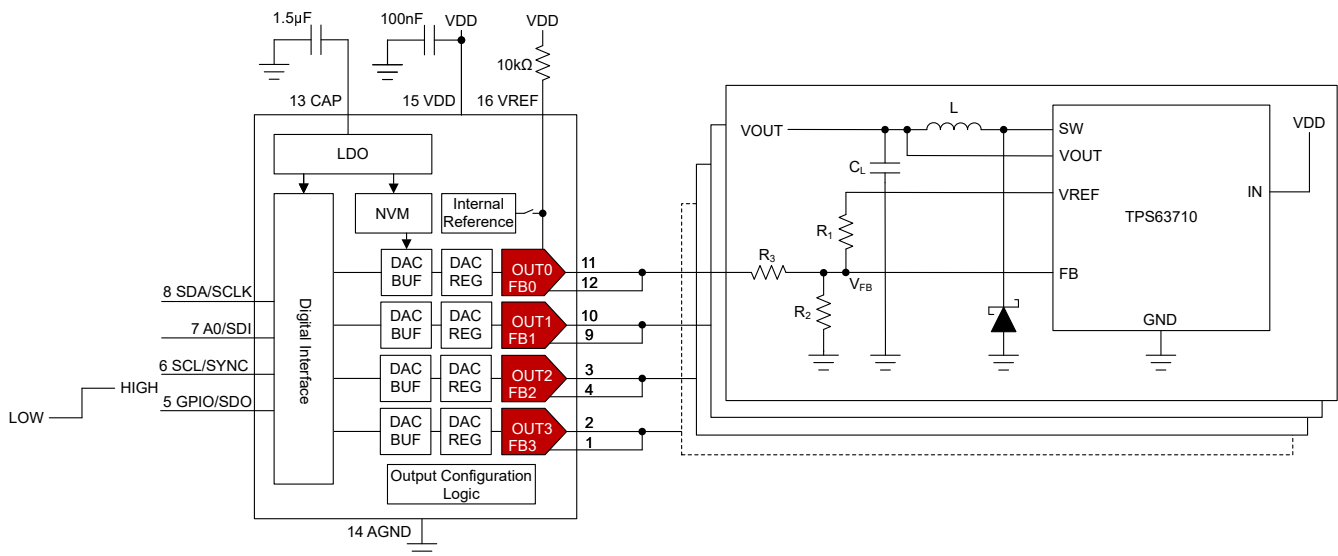
### 设计目标

主要输入参数	主要输出信号	推荐器件
通过 SPI 或 I <sup>2</sup> C 通信控制 DAC 电压输出	0V 至 1.2V 可编程灌电流， $-3.45V \pm 26%$ 直流/ 直流输出	DAC63204W、DAC53204W、DAC63004W、 DAC53004W、DAC63204、DAC53204、DAC43204、 TPS63710

目标：将直流/直流输出裕度设置为标称值的  $\pm 26%$ 。

### 设计说明

此电路使用四通道缓冲电压输出 DAC 对反相直流/直流降压转换器进行裕度调节。电源裕量调节电路用于修整、调节或测试电源转换器的输出。低压降稳压器 (LDO)、直流/直流转换器或开关模式电源 (SMPS) 等可调节电源提供反馈 (FB) 引脚，该引脚与电阻分压器一起用于控制所需的输出。DAC63204W 或 DAC53204W (DACx3204W) 等精密智能 DAC 通过使用串联电阻器将电流注入分压器，从而实现对电源输出的线性控制。DACx3204 有一个通用输入 (GPI) 引脚，此引脚可使 DAC 输出在高输出电压和低输出电压之间切换。这样可在标称输出值的  $\pm 26%$  范围内切换直流/直流转换器。所有寄存器设置均使用集成的非易失性存储器 (NVM) 进行保存，即使在下电上电或复位后，也能在没有运行时软件的情况下使用器件。此电路可用于激光雷达、虚拟现实耳机和 OLED 电视等应用。



## 设计说明

1. **DACx3204W** 采用 **DSBGA** 封装且带有自动检测型 **I2C**、**SPI** 或 **PMBus®** 接口的 **12 位** 和 **10 位**、**四通道电压和电流输出智能 DAC** 数据表建议为 **VDD** 引脚使用 **100nF** 去耦电容器，为 **CAP** 引脚使用 **1.5μF** 或更大的旁路电容器。**CAP** 引脚连接至内部 **LDO**。将这些电容器靠近器件引脚放置。
2. 如果使用了外部基准，则在 **VREF** 引脚与 **GND** 之间连接一个 **100nF** 电容器。在 **VDD** 之后斜升外部基准。如果未使用外部基准，则在 **VREF** 引脚与 **VDD** 之间连接一个上拉电阻器。此示例使用内部基准，**VREF** 引脚通过 **10kΩ** 电阻器上拉至 **VDD**。
3. 未连接 **DAC63204W** 或通过串联电阻器 **R<sub>3</sub>** 的电流为 **0A** 时，**TPS63710** 的输出电压 (**V<sub>OUT</sub>**) 通过电阻器 **R<sub>1</sub>** 和 **R<sub>2</sub>** 来设置。**TPS63710** 在 **FB** 引脚使用内部 **-700mV** 基准电压 (**V<sub>FB</sub>**) 来确定 **V<sub>OUT</sub>**。当 **DAC63204W** 输出电压 (**V<sub>DAC</sub>**) 等于 **V<sub>FB</sub>** 时，流经 **R<sub>3</sub>** 的电流为 **0A**。**DAC63204W** 无法输出负电压，因此该设计假设 **DAC63204W** 输出始终为正，并且当 **DAC63204W** 处于中标度时，**TPS63710** 处于标称输出电压。
4. 选择 **R<sub>3</sub>**，以便当 **DAC63204W** 设置为断电模式时 **V<sub>DAC</sub> > -0.3V**。当 **DAC63204W** 配置为 **10kΩ** 至 **GND** 断电模式时，**10kΩ** 电阻会通过 **R<sub>3</sub>** 创建一个电阻分压器。在该示例中，**R<sub>3</sub>** 选择为 **200kΩ**，从而在处于 **10kΩ** 至 **GND** 断电模式时使 **V<sub>DAC</sub>** 等于 **-0.038V**。当负电压连接到 **V<sub>DAC</sub>** 时，请勿使用高阻态断电模式。
5. 选择流经 **R<sub>2</sub>** (**I<sub>R2</sub>**) 的电流，这样流入 **TPS63710 FB** 引脚的偏置电流可以忽略不计。使用以下公式计算 **R<sub>2</sub>**：

$$R_2 = \frac{V_{FB}}{I_{R2}}$$

**I<sub>R2</sub>** 选为 **5.2μA**。**TPS63701** 的内部增益系数为 **1/0.9**，这使得有效 **V<sub>FB</sub>** 为 **-778mV**。**R<sub>2</sub>** 的计算公式为：

$$R_2 = \frac{|-778 \text{ mV}|}{5.2 \text{ μA}} = 150 \text{ k}\Omega$$

6. 当 **V<sub>DAC</sub>** 处于中标度或 **0.91V** 时，标称 **TPS63710 V<sub>OUT</sub>** 选择为 **-3.45V**。可以通过以下公式来计算 **DAC63204W** 输出提供的电流：

$$I_{DAC} = \frac{V_{DAC} - V_{FB}}{R_3}$$

$$I_{DAC} = \frac{910 \text{ mV} + 778 \text{ mV}}{200 \text{ k}\Omega} = 8.44 \text{ μA}$$

可以使用以下公式来计算 **R<sub>1</sub>**，以实现期望的标称 **V<sub>OUT</sub>**：

$$R_1 = \frac{V_{FB} - V_{OUT}}{I_{R2} - I_{DAC}}$$

$$R_1 = \frac{-0.778 \text{ V} + 3.45 \text{ V}}{5.2 \text{ μA} + 8.44 \text{ μA}} = 196 \text{ k}\Omega$$

7. **DAC63204W** 通过调节 **V<sub>DAC</sub>**，通过 **R<sub>1</sub>** 灌入或拉取额外电流，从而实现所需的裕度。**V<sub>DAC</sub>** 的计算公式为：

$$V_{DAC} = \left( I_{R2} - \frac{V_{OUT} - V_{FB}}{R_1} \right) \times R_3 + V_{FB}$$

**V<sub>DAC,MAX</sub>** 和 **V<sub>DAC,MIN</sub>** 配置为将 **V<sub>OUT</sub>** 裕度增加 **26%**。**V<sub>OUT</sub>** 低电平为 **-4.34V**，**V<sub>OUT</sub>** 高电平为 **-2.55V**

$$V_{DAC,MAX} = \left( -5.2 \text{ μA} - \frac{-4.34 \text{ V} + 0.778 \text{ V}}{196 \text{ k}\Omega} \right) \times 200 \text{ k}\Omega - 0.778 \text{ V} = 1.82 \text{ V}$$

$$V_{DAC,MIN} = \left( -5.2 \text{ μA} - \frac{-2.55 \text{ V} + 0.778 \text{ V}}{196 \text{ k}\Omega} \right) \times 200 \text{ k}\Omega - 0.778 \text{ V} = 0 \text{ V}$$

8.  $V_{DAC,MAX}$  和  $V_{DAC,MIN}$  的 DAC 代码存储在 DAC-X-MARGIN-HIGH 和 DAC-X-MARGIN-LOW 寄存器中。  
 $V_{DAC,NOM}$  存储在 DAC-X-DATA 寄存器中。使用以下公式计算以十进制形式编程到这些寄存器的代码：

$$DAC\_CODE = \frac{V_{DAC} \times 2^{12}}{V_{REF}}$$

此设计使用内部 1.21V 基准，增益为  $\times 1.5$ ，满量程电压为 1.82V。公式变为：

$$DAC\_MARGIN\_HIGH = \frac{1.82\text{ V} \times 2^{12}}{1.82\text{ V}} = 4096\text{d}$$

$$DAC\_DATA = \frac{0.91\text{ V} \times 2^{12}}{1.82\text{ V}} = 2048\text{d}$$

$$DAC\_MARGIN\_LOW = \frac{0\text{ V} \times 2^{12}}{1.82\text{ V}} = 0\text{d}$$

12 位器件的最大输出代码为 4095，因此  $V_{DAC,MAX}$  变为 1.819V。

9. TPS63710 要求  $V_{IN} \geq |V_{OUT}|/0.7$ 。此应用的最大  $V_{OUT}$  为  $-4.34\text{V}$ ，因此最小  $V_{IN}$  为 6.2V。此设计中使用 10V。
10. 使用具有  $\times 1.5$  增益的 1.21V 基准和 12 位 DAC63204W，每个代码之间的 LSB 大小或步长大约为  $443\mu\text{V}$ 。使用尽可能低的基准电压可减小 LSB 大小，从而更大程度地提高  $V_{DAC,MAX}$  和  $V_{DAC,MIN}$  的分辨率。
11. DAC63204W 具有可编程压摆率特性。通过 DAC-X-FUNC-CONFIG 寄存器中的 CODE-STEP-X 和 SLEW-RATE-X 字段配置可编程压摆率。仅当在 DAC-X-MARGIN-HIGH 和 DAC-X-MARGIN-LOW 寄存器中存储的两个值之间切换时，才可使用可编程压摆率。

CODE-STEP-X 定义了用于从起始代码过渡到最终输出代码的 LSB 步进数。SLEW-RATE-X 定义了每个代码步长的时间周期。压摆时间的计算公式为：

$$t_{SLEW} = SLEW\_RATE \times \text{CEILING}\left(\frac{MARGIN\_HIGH\_CODE - MARGIN\_LOW\_CODE}{CODE\_STEP} + 1\right)$$

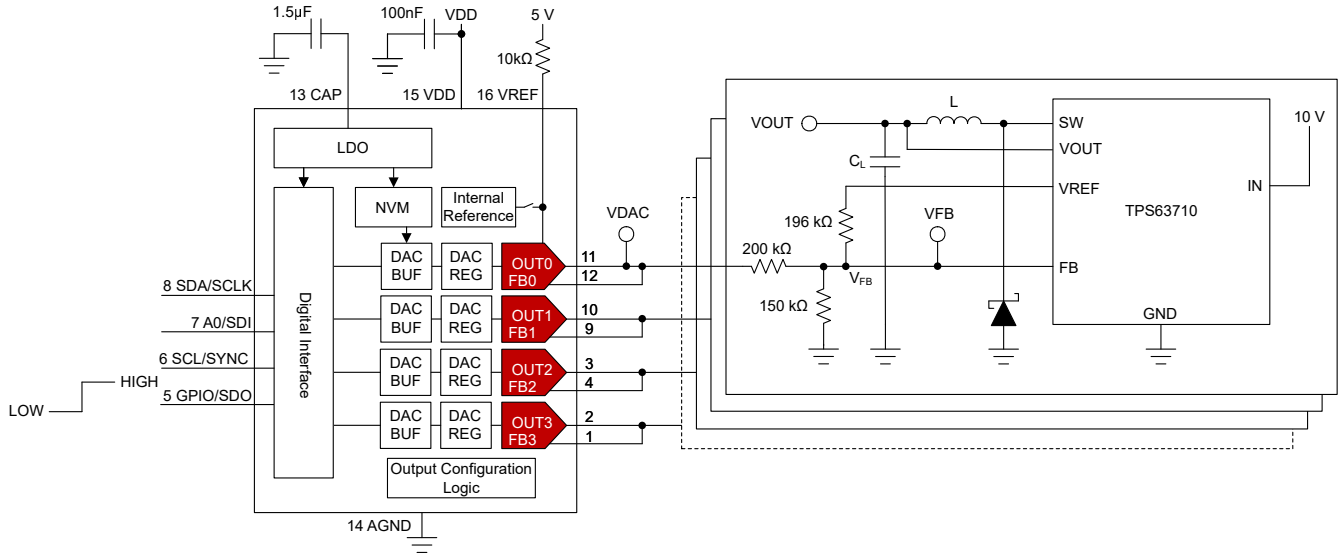
此应用使用裕度高代码 4095、裕度低代码 0、SLEW-RATE 为  $1282\mu\text{s}/\text{LSB}$ ，以及 CODE-STEP 为 1 LSB 来实现 5.25s 的压摆时间：

$$t_{SLEW} = 1282\left(\mu\text{s}/\text{LSB}\right) \times \text{CEILING}\left(\frac{4095 - 0}{1\text{ LSB}} + 1\right) = 5.25\text{ s}$$

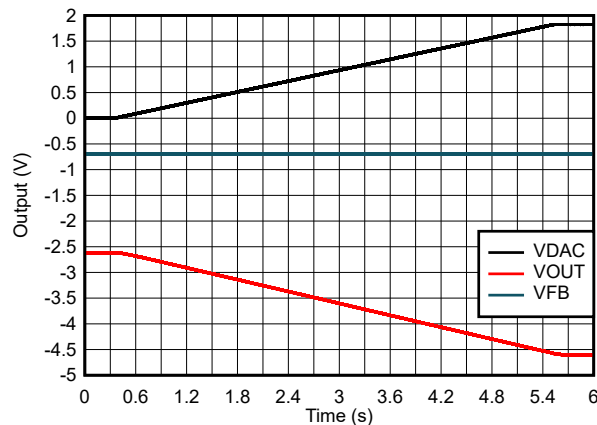
12. GPIO 引脚可配置为数字输入，以便通过可编程压摆率在裕度高输出和裕度低输出之间切换 DAC63204W 输出。GPIO-CONFIG 寄存器中的 GPI-EN 位使 GPIO 引脚作为输入。GPI-CH-SEL 字段选择由 GPI 控制的通道。GPI-CONFIG 字段选择 GPI 功能。将 0b1010 写入 GPI-CONFIG 字段，以配置 GPIO 引脚，从而触发高裕度或低裕度功能。
- GPI 上的高电平将 DAC 输出设置为  $V_{DAC,MAX}$ ，并将 TPS63710  $V_{OUT}$  设置为低电平或  $-4.34\text{V}$ 。GPI 上的低电平将 DAC 输出设置为  $V_{DAC,MIN}$ ，并将 TPS63710  $V_{OUT}$  设置为高电平或  $-2.55\text{V}$ 。
13. 使用 I<sup>2</sup>C 或 SPI，根据寄存器设置部分所述的初始寄存器设置对 DAC63204W 进行编程。将初始寄存器设置保存在 NVM 中，方法是将 1 写入 COMMON-TRIGGER 寄存器的 NVM-PROG 字段。对 NVM 进行编程后，器件将在重置或下电上电之后加载具有 NVM 所存储值的所有寄存器。

## 设计结果

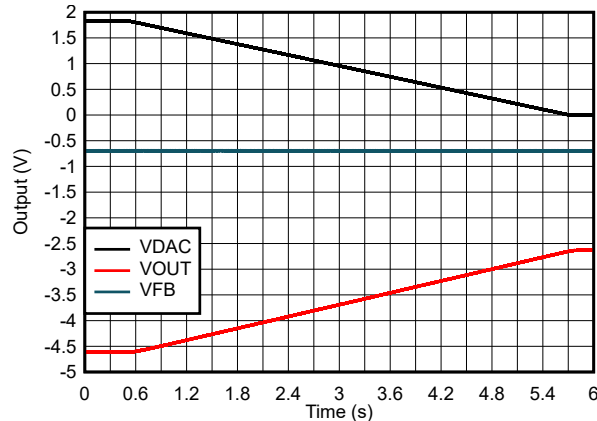
此原理图用于 DAC63204W 的以下设计结果。根据原理图上标记的测试点，在示波器上测量  $V_{DAC}$ 、 $V_{OUT}$  和  $V_{FB}$  信号。



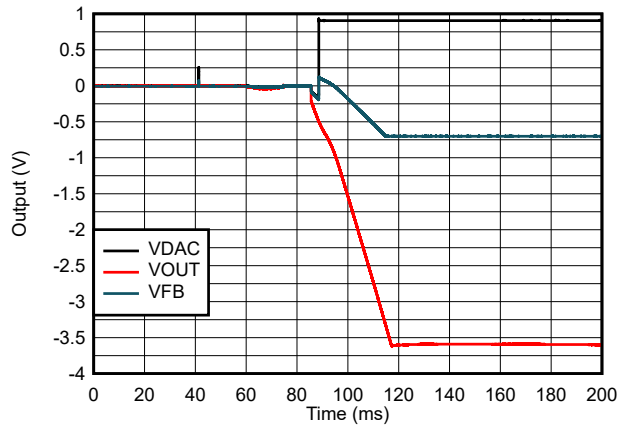
此图显示了在使用设计说明中讨论的设置配置 5.25s 压摆时间后，DAC63204W 输出从低电平到高电平的转换。 $V_{DAC}$  输出电压从 0V 转换为 1.82V，这会导致 TPS63710  $V_{OUT}$  电压从 -2.55V 转换为 -4.34V。



此图显示了 DAC63204W 输出在 5.25s 压摆时间下从低电平到高电平的转换。 $V_{DAC}$  输出电压从 1.82V 转换为 0V，这会导致 TPS63710  $V_{OUT}$  电压从 -4.34V 转换为 -2.55V。



此图显示了电路的启动行为。10V TPS63710 电源和 5V DAC63204W 电源同时打开。 $V_{DAC}$  输出启动至 910mV 的标称电压。施加  $V_{FB}$  基准电压时，TPS63710  $V_{OUT}$  斜升至 -3.45V 的标称输出。



### 寄存器设置

下表显示了此应用的示例寄存器映射。此处给出的值适用于在设计说明部分作出的设计选择。

**DAC63204W 的寄存器设置**

寄存器地址	寄存器名称	设置	说明
0x1F	COMMON-CONFIG	0x1249	[15] 0b0 : 写入 0b0, 将窗口比较器输出设置为非锁存输出
			[14] 0b0 : 器件未锁定
			[13] 0b0 : 故障转储读取使能位于地址 0x00 处
			[12] 0b1 : 启用内部基准
			[11:10] 0b00 : 为 VOUT3 上电
			[9] 0b1 : 将 IOOUT3 断电
			[8:7] 0b00 : 为 VOUT2 上电
			[6] 0b1 : 将 IOOUT2 断电
			[5:4] 0b00 : 为 VOUT1 上电
			[3] 0b1 : 将 IOOUT1 断电
			[2:1] 0b00 : 为 VOUT0 上电
[0] 0b1 : 将 IOOUT0 断电			
0x24	GPIO-CONFIG	0x01F5	[15] 0b0 : 为 GP 输入禁用干扰滤波器
			[14] 0b0 : 不用考虑
			[13] 0b0 : 禁用 GPIO 引脚的输出模式
			[12:9] 0b0000 : 选择映射到 GPIO 作为输出的 STATUS 功能设置
			[8:5] 0b1111 : 在所有通道上启用 GPI 功能
			[4:1] 0b1010 : GP 输入配置为触发裕度高或裕度低
			[0] 0b1 : 启用 GPIO 引脚的输入模式
0x20	COMMON-TRIGGER	0x0002	[15:12] 0b0000 : 写入 0b0101, 解锁器件
			[11:8] 0b0000 : 写入 0b1010, 触发 POR 复位
			[7] 0b0 : 不触发 LDAC
			[6] 0b0 : 不触发 DAC 清除
			[5] 0b0 : 不用考虑
			[4] 0b0 : 不触发故障转储
			[3] 0b0 : 不触发 PROTECT 功能
			[2] 0b0 : 不触发故障转储读取
			[1] 0b1 : 写入 0b1, 将适用寄存器设置存储到 NVM
			[0] 0b0 : 不触发 NVM 重新加载。写入 0b1, 使用现有 NVM 设置重新加载适用寄存器

**DAC63204W 的寄存器设置 (continued)**

寄存器地址	寄存器名称	设置	说明
0x03、0x09、 0x0F、0x15	DAC-X-VOUT-CMP- CONFIG	0x0800	[15:13] 0b000 : 不用考虑
			[12:10] 0b010 : 选择具有 $\times 1.5$ 增益的内部基准
			[9:5] 0x00 : 不用考虑
			[4] 0b0 : 在比较器模式中将 OUTx 引脚设置为推挽
			[3] 0b0 : 在内部消耗的比较器输出
			[2] 0b0 : FBx 输入在比较器模式下具有高阻抗
			[1] 0b0 : 不反转比较器输出
			[0] 0b0 : 禁用比较器模式
0x06、0x0C、 0x12、0x18	DAC-X-FUNC-CONFIG	0x000D	[15] 0b0 : DAC-X 清除模式设置为零标度
			[14] 0b0 : DAC-X 输出在写命令后立即更新
			[13] 0b0 : 不使用广播命令更新 DAC-X
			[12:11] 0b00 : 相位设置为 0°
			[10:8] 0b000 : 选择正弦波模式
			[7] 0b0 : 启用线性转换
			[6:4] 0b000 : 选择 8 LSB 代码步长
			[3:0] 0xD : 选择 8 $\mu$ s/步进的压摆率
0x01、0x07、 0x0D、0x13	DAC-X-MARGIN-HIGH	0xFFFF0	[15:4] 0xFFFF : 12 位裕度高代码
			[3:0] 0x0 : 不用考虑
0x02、0x08、 0x0E、0x14	DAC-X-MARGIN-LOW	0x0000	[15:4] 0x000 : 12 位裕度低代码
			[3:0] 0x0 : 不用考虑

## 伪代码示例

下面所示为将初始寄存器值编程到 DAC63204W 的 NVM 的伪代码序列。此处给出的值适用于在 [设计说明](#) 部分作出的设计选择。

### DAC63204W 伪代码示例

```
1: //SYNTAX: WRITE <REGISTER NAME (Hex code)>, <MSB DATA>, <LSB DATA>
2: //Set gain setting to 1.5x internal reference (1.8 V) (repeat for all channels)
3: WRITE DAC-0-VOUT-CMP-CONFIG(0x3), 0x08, 0x00
4: //Power-up voltage output on all channels and enable the internal reference
5: WRITE COMMON-CONFIG(0x1F), 0x12, 0x49
6: //Configure GPI for Margin-High, Low trigger for all channels
7: WRITE GPIO-CONFIG(0x24), 0x01, 0xF5
8: //Set slew rate and code step (repeat for all channels)
9: //CODE_STEP: 1 LSB, SLEW_RATE: 1282 µs/step
10: WRITE DAC-0-FUNC-CONFIG(0x06), 0x00, 0x0D
11: //write nominal DAC code (repeat for all channels)
12: //For a 1.8-V output range, the 12-bit hex code for 0.9 V is 0x800. With 16-bit left alignment,
13: this becomes 0x8000
14: WRITE DAC-0-DATA(0x19), 0x80, 0x00
15: //Write DAC margin high code (repeat for all channels)
16: //For a 1.8-V output range, the 12-bit hex code for 1.8 V is 0xFFF. With 16-bit left alignment,
17: this becomes 0xFFFF0
18: WRITE DAC-0-MARGIN-HIGH(0x01), 0xFF, 0xF0
19: //Write DAC margin low code (repeat for all channels)
20: //The 12-bit hex code for 0 V is 0x000. With 16-bit left alignment, this becomes 0x0000
21: WRITE DAC-0-MARGIN-LOW(0x02), 0x00, 0x00
21: //Save settings to NVM
22: WRITE COMMON-TRIGGER(0x20), 0x00, 0x02
```

## 设计中采用的器件

器件	关键特性	链接
DAC63204W	采用 DSBGA 封装、具有 I <sup>2</sup> C、SPI 和在断电期间处于高阻态输出的 4 通道、12 位 VOUT 和 IOOUT 智能 DAC	<a href="#">DAC63204W</a>
DAC53204W	采用 DSBGA 封装、具有 I <sup>2</sup> C、SPI 和在断电期间处于高阻态输出的 4 通道、10 位 VOUT 和 IOOUT 智能 DAC	<a href="#">DAC53204W</a>
DAC63004W	采用 DSBGA 封装、具有 I <sup>2</sup> C、SPI 和在断电期间处于高阻态输出的 4 通道、12 位 VOUT 和 IOOUT 智能 DAC	<a href="#">DAC63004W</a>
DAC53004W	采用 DSBGA 封装、具有 I <sup>2</sup> C、SPI 和在断电期间处于高阻态输出的 4 通道、10 位 VOUT 和 IOOUT 智能 DAC	<a href="#">DAC53004W</a>
TPS63710	采用 3x3 WSON 封装的低噪声 1 A 同步反相降压转换器	<a href="#">TPS63710</a>

使用 [参数搜索工具](#) 查找其他可能的器件。

## 设计参考资料

有关 TI 综合电路库的信息，请参阅 [模拟工程师电路手册](#)。

## 附加资源

- 德州仪器 (TI), [DAC63004WCSP 评估模块](#)
- 德州仪器 (TI), [DAC63004WCSP-EVM 用户指南](#)
- 德州仪器 (TI), [精密实验室 - DAC](#)
- 德州仪器 (TI), [TPS63710EVM-811](#)

如需 TI 工程师的直接支持，请使用 E2E 社区：

[e2e.ti.com](http://e2e.ti.com)



## 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