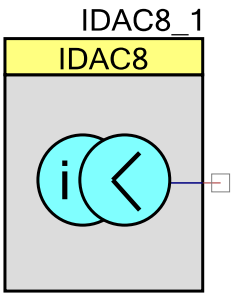
8 位电流数模转换器 (IDAC8)

1.60



特性

- 三个电流范围：2040 µA、255 µA 和 32.875 µA
- 灌电流或源电流可选
- 软件或时钟驱动输出探针
- 数据源可以为 CPU、DMA 或 UDB

概述

IDAC8 组件是 8 位电流输出 DAC（数模转换器）。输出的源电流或灌电流具有三个范围。IDAC8 可通过硬件、软件或两者相结合来进行控制。

输入/输出连接

本节介绍 IDAC8 的各种输入和输出连接。I/O 列表中的星号 (*) 表示该 I/O 可能在 I/O 说明中列出的情况下隐藏在符号中。

Iout – 模拟

Iout 终端用于连接源电流/灌电流。可以路由至设备上任何兼容模拟的引脚上。选定最高电流范围 (2048 µA) 时，输出只能路由至一组提供直流低阻通路的指定引脚上。这些端口引脚是 P0[6]、P0[7]、P3[0] 或 P3[1]。

data[7:0] – 输出 *

此 8 位宽数据信号将 IDAC8 直接连接到 DAC 总线。DAC 总线可以由基于 UDB 的组件或控制寄存器驱动，或者它也可以从 GPIO 引脚直接路由。可以通过将 **Data_Source** 参数设置为“DAC 总线”来启用此输入。如果选择了“DMA 或 CPU”选项，则总线连接将从组件符号中消失。

当硬件能够在没有 CPU 干预的情况下设置正确的值时，使用 data[7:0] 输入。使用此选项时，还应启用 **Strobe_Mode**。

对于许多应用场合，不需要此输入，但是 CPU 或 DMA 会将值直接写入数据寄存器。在固件方

面，可使用 SetRange()函数或将值直接写入 IDAC8_1_Data 寄存器中（假设实例名称为“IDAC8_1”）。

探针 – 输入 *

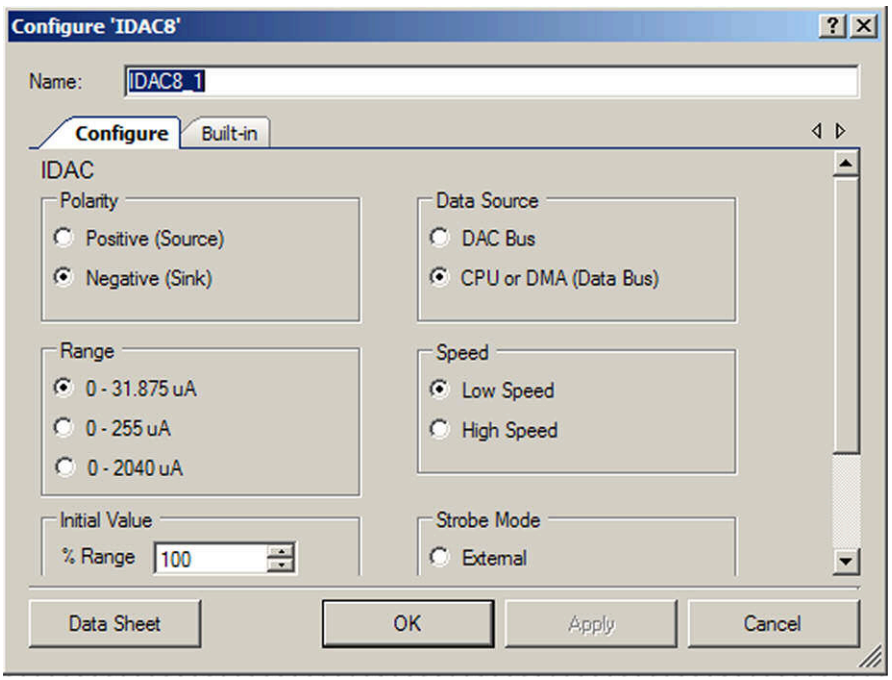
探针输入是可选信号输入，是通过 **Strobe_Mode** 参数选择的。 如果 **Strobe_Mode** 设置为 “外部”，则引脚将是可见的，且必须连接到有效数据源。在此模式中，数据从 IDAC8 寄存器传输到探针信号下一个正边沿上的 DAC。

如果此参数设置为“寄存器写入”，则引脚将从符号中消失，对数据寄存器的任何写入将立即传输到 DAC。

对于音频或定期采样应用场合，还可以使用对传入 DAC 的数据进行测速的同一时钟来生成中断。时钟的每个上升沿都将数据传输到 DAC，导致中断获取加载到 DAC 寄存器的下一个值。

参数和设置

将 IDAC8 组件拖放到您的设计上，然后双击以打开“配置”对话框。



IDAC8 组件提供下列参数。

Data_Source

此参数选择要写入 DAC 寄存器的数据源。如果 CPU（固件）或 DMA 将数据写入 IDAC8，则选择“DMA 或 CPU”。如果通过 UDB 或基于 UDB 的组件直接写入数据，则选择“DAC 总线”。请注意，仅有一条 DAC 总线由所有 VIDAC8 模拟模块共享。

IDAC_Range

通过此参数，设计者能够将三个电流范围中的一个设为默认值。在运行时可以随时使用 SetRange() 函数更改范围。如果选定最高电流范围：“0 - 2040uA”，输出应路由至提供低阻通路的指定引脚之一上。这些引脚是 P0[6]、P0[7]、P3[0] 和 P3[1]。

范围	最低值	最高值	步进大小
0 – 32 uA	0.0 µA	31.875 µA	0.125 µA
0 – 255 uA	0.0 µA	255 µA	1 µA
0 – 2040 uA	0.0 µA	2040 µA	8 µsA

IDAC_Speed

此参数为设计者提供了两种设置：“低速”和“高速”。在“低速”模式下，建立时间较慢，但是使用的工作电流较小。在“高速”模式下，电流处理速度快得多，但是代价是工作电流较大。

Initial_Value

这是执行 Start() 命令后 IDAC8 将提供的初始值。SetValue() 函数或对 DAC 寄存器的直接写入将随时覆盖默认值。合法值介于 0 和 255 之间（含）。

极性

通过“极性”参数，设计者能够选择用 IDAC8 对负载施加灌电流还是源电流。选择“源电流”选项时，DAC 输出将向负载施加源电流，负载连接的电压为 Vss 或其他比 Vdda 至少低 1.0 V 的电压。在“灌电流”模式中，DAC 输出将向连接至 Vdd 或其他比至少 Vss 高 1.0 V 的电压的负载提供电流。选择“极性”时，符号表示电流的方向。

Strobe_Mode

此参数选择当数据写入 IDAC8 数据寄存器时数据是否立即写入 DAC。当选择“寄存器写入”选项时，选择此模式。当选择“外部”选项时，来自 UDB 的时钟或信号控制数据何时从 DAC 寄存器写入实际 DAC。

资源

模拟模块	数字模块					API 存储器（字节）		引脚（每个外部 I/O）
	数据路径	宏单元	状态寄存器	控制寄存器	计数器 7	闪存	RAM	
1 个 VIDAC 固定模块	不可用	不可用	不可用	不可用	不可用	417	3	1

IDAC8 使用一个 VIDAC8 模拟模块。

应用程序编程接口

应用程序编程接口 (API) 子程序允许您使用软件配置组件。下表列出了每个函数的接口，并进行了说明。以下各节将更详细地介绍每个函数。

默认情况下，PSoC Creator 将实例名称“IDAC8_1”分配给指定设计中组件的第一个实例。您可以将该实例重命名为符合标识符语法规则的任意唯一值。实例名称会成为每个全局函数名称、变量和常量符号的前缀。出于可读性考虑，下表中使用的实例名称为“IDAC8”。

函数	说明
void IDAC8_Start(void)	使用默认自定义程序值初始化 IDAC8。启用 IDAC8 并加电。
void IDAC8_Stop(void)	禁用 IDAC8 并将其设置为最低功耗状态。
Void IDAC8_SetSpeed(uint8 speed)	设置 DAC 速度。
void IDAC8_SetPolarity(uint8 polarity)	将输出模式设为灌电流或源电流。
void IDAC8_SetRange(uint8 range)	设置 IDAC8 的全量程范围。
void IDAC8_SetValue(uint8 value)	使用给定范围设置 0 到 255 之间的值。
void IDAC8_SaveConfig(void)	空函数。提供以便将来使用
void IDAC8_RestoreConfig(void)	空函数。提供以便将来使用。
void IDAC8_Sleep(void)	停止并保存用户配置。
void IDAC8_WakeUp(void)	恢复并启用用户配置。
void IDAC8_Init(void)	初始化或恢复默认 IDAC8 配置
void IDAC8_Enable(void)	启用 IDAC8。

全局变量

变量	说明
IDAC8_initVar	指示 IDAC8 是否已初始化。变量初始化为 0，并在第一次调用 IDAC8_Start() 时设置为 1。这样，第一次调用 IDAC8_Start() 子程序后，组件不用重新初始化即可重启。 如需重新初始化组件，可在 IDAC8_Start() 或 IDAC8_Enable() 函数前调用 IDAC8_Init() 函数。

void IDAC8_Start(void)

说明： 这是开始执行组件操作的首选方法。IDAC8_Start() 设置 initVar 变量，调用 IDAC8_Init() 函数，然后调用 IDAC8_Enable() 函数。启用 IDAC8，并加电至给定的功耗水平。功耗水平 0 相当于执行停止函数。

参数： 无

返回值： 无

副作用： 如果已设置 initVar 变量，则该函数仅调用 _Enable() 函数。

void IDAC8_Stop(void)

说明： 将 IDAC8 断电至最低功耗状态，并禁用输出。

参数： 无

返回值： 无

副作用： 无

void IDAC8_SetSpeed(uint8 speed)

说明： 设置 DAC 速度。

参数： (uint8) speed：设置 DAC 速度，参见下表可获取有效参数。

选项	说明
IDAC8_LOWSPEED	低速（低功耗）
IDAC8_HIGHSPEED	高速（高功耗）

返回值： 无

副作用： 无

void IDAC8_SetPolarity(uint8 polarity)

说明： 将输出极性设为“灌”或“源”。

参数： (uint8) polarity：设置灌电流或源电流功能，请参阅下表。

选项	说明
IDAC8_SOURCE	设置为“源电流”模式。
IDAC8_SINK	设置为“灌电流”模式。

返回值： 无

副作用： 无

void IDAC8_SetRange(uint8 range)

说明： 设置 IDAC8 的全量程范围

参数： (uint8) range：设置 IDAC8 的全量程范围。有关范围，请参见下表。

选项	说明
IDAC8_RANGE_32uA	将全量程范围设为 31.875 µA
IDAC8_RANGE_255uA	将全量程范围设为 255 µA
IDAC8_RANGE_2mA	将全量程范围设为 2.040 mA

返回值： 无

副作用： 无

void IDAC8_SetValue(uint8 value)

说明： 将值设置为 IDAC8 上的输出。有效值介于 0 和 255 之间。

参数： (uint8) value：介于 0 和 255 之间的值。值 0 是最低值（零），值 255 是全量程值。全量程值取决于可使用 SetRange API 选择的范围。

返回值： 无

副作用： 在 PSoC 3 ES2 和 PSoC 5 ES1 上，应当在启用 IDAC 电源后调用 IDAC8_SetValue() 函数。

void IDAC8_Sleep(void)

说明： 这是准备组件进入睡眠的首选 API。IDAC8_Sleep() API 保存当前组件状态。然后调用 IDAC8_Stop() 函数，并调用 IDAC8_SaveConfig() 以保存硬件配置。

在调用 CyPmSleep() 或 CyPmHibernate() 函数之前调用 IDAC_Sleep() 函数。有关电源管理功能的更多信息，请参考 PSoC Creator 的 System Reference Guide（《系统参考指南》）。

参数： 无

返回值： 无

副作用： 无

void IDAC8_Wakeup(void)

说明： 这是将组件恢复为调用 IDAC8_Sleep() 时的状态的首选 API。IDAC8_Wakeup() 函数调用 IDAC8_RestoreConfig() 函数以恢复配置。如果组件在调用 IDAC8_Sleep() 函数前已启用，则 IDAC_Wakeup() 函数也将重新启用组件。

参数： 无

返回值： 无

副作用： 调用 IDAC8_Wakeup() 函数前未调用 IDAC8_Sleep() 或 IDAC8_SaveConfig() 函数可能会产生意外行为。

void IDAC8_SaveConfig(void)

说明： 此函数保存组件配置。它将保存非保留寄存器。此函数还将保存当前“配置”对话框中定义的或通过相应 API 修改的组件参数值。此函数由 IDAC8_Sleep() 函数调用。

参数： 无

返回值： 无

副作用： 空函数。实现以供将来使用。调用此函数对组件没有作用。

void IDAC8_RestoreConfig(void)

说明： 此函数恢复组件配置。此将恢复非保留寄存器。此函数还将组件参数值恢复为调用 IDAC8_Sleep() 函数之前的值。

参数： 无

返回值： 无

副作用： 空函数。实现以供将来使用。调用此函数对组件没有作用。

void IDAC8_Init(void)

说明： 根据自定义程序“配置”对话框设置来初始化或恢复组件。不需要调用 IDAC8_Init()，因为 IDAC8_Start() API 会调用此函数，这是开始组件操作的首选方法。

参数： 无

返回值： 无

副作用： 所有寄存器将根据自定义程序“配置”对话框设置为相应的值。这将重新初始化组件。如果您要设置除寄存器中当前值之外的新值，则调用 IDAC8_Init() 函数需要调用 IDAC8_SetValue()。

void IDAC8_Enable(void)

说明： 激活硬件并开始执行组件操作。不需要调用 IDAC8_Enable()，因为 IDAC8_Start() API 会调用此函数，这是开始组件操作的首选方法。

参数： 无

返回值： 无

副作用： 无

固件源代码示例

PSoC Creator 在“查找示例项目”对话框中提供了大量包括原理图和代码示例的示例项目。要获取组件特定的示例，请打开组件目录中的对话框或原理图中的组件实例。要获取通用的示例，请打开开始页或**文件**菜单中的对话框。 根据需要，使用对话框中的**滤波器选项**可缩小可选项目的列表。

有关更多信息，请参考 PSoC Creator 帮助中的“查找示例项目”主题。

功能描述

DAC 可生成电压或电流输出。DAC 采用电流镜结构；电流从参考源映射到镜像 DAC 处。校准电流镜和数值电流镜负责 8 位校准 [DACx_TR] 和 8 位 DAC 值。然后，电流转入分频器中，生成对应于 DAC 值的电流。DAC 值可取自寄存器 DACx_D，或者也可取自 UDB 的 8 条线路。通过 DACx_CR1[5] 位选择此选项。

DAC 具有探针，可获得其输出，以更改输入代码。通过 DACx_STROBE[3] 位启用探针控制。可从总线写入探针、模拟时钟探针和任何 UDB 信号探针中选择 DAC 的探针源。在 DACx_STROBE[2:0] 设置的基础上完成该选项。

电流 (IDAC) 模式

源电流和灌电流的两个电流镜分别为其提供输出，作为源电流和灌电流。在电流模式下，这些电流镜也提供范围选项。

用作 IDAC 时，输出为 8 位数模转换电流。通过设置 DACx_CR0 [4] 寄存器实现此目标。参考源为来自称为 IREF(DAC) 的模拟参考的电流参考。在此模式中，可通过寄存器 DACx_CR0 [3:2] 选择三种输出范围：

- 0 至 2.048 mA，8 µA/bit
- 0 至 256 µA，1 µA/bit
- 0 至 32 µA，0.125 µA/bit

对于各个级别，都有 255 个相等的步进，步长为 M/256，其中 M = 2.048mA、256 µA 或 32 µA。在 2.040-mA 配置中，模块旨在将电流输出至外部 600Ω 负载中。输出可能会递交至任何电阻或固定电压，只要其满足 1.0 V 的最小压差要求。这意味着，源电流的最大电压为 Vdda – 1.0V，灌电流的最小输出电压为 Vssa + 1.0V。

IDAC 转换速率能够达到 8 Msps。您也可以选择输出是源电流还是灌电流。通过 DACx_CR1[2] 寄存器实现此目标。也可用 UDB 输入，完成对 IDAC 源电流和灌电流的选择。可使用 DACx_CR1[3] 位启用源电流和灌电流选择的 UDB 控制。

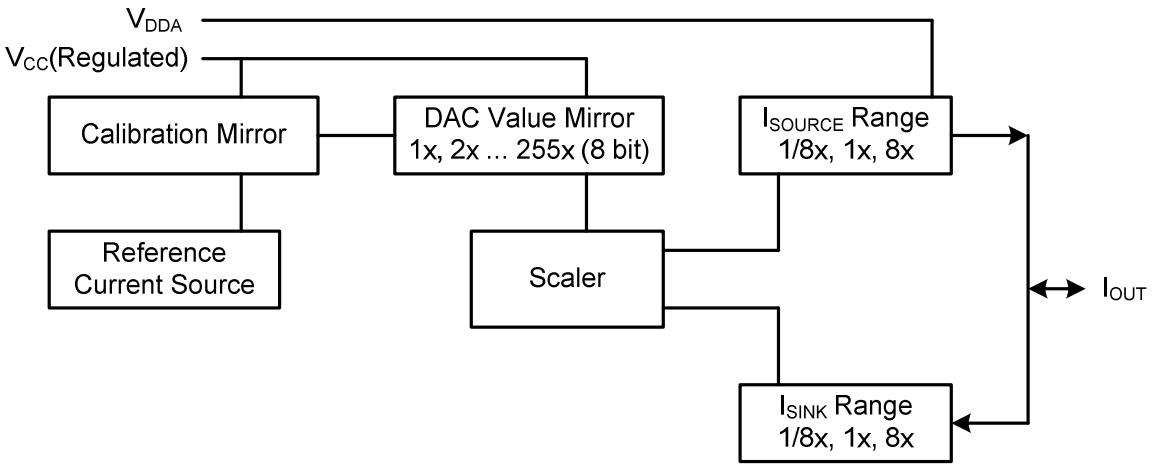
DMA

IDAC8 组件不需要实现 DMA 请求信号。典型的用法是信号生成，应当在外部控制传输到 IDAC8 组件的数据速率。可以使用 DMA 向导按如下所示配置 DMA 操作：

DMA 向导中 DMA 源/目标的名称	方向	DMA 请求信号	DMA 请求类型	说明
IDAC8_Data_PTR	目标	不可用	不可用	存储介于 0 到 255 的 DAC 值

框图和配置

下面显示了 IDAC8 组件的框图。



寄存器

提供的函数支持大多数应用场合下所需的大多数通用运行时函数。以下寄存器参考信息为高级用户提供了简要的说明。无需使用 API，可以使用 IDAC8_Data 寄存器将数据直接写入 DAC。这对于 CPU 或 DMA 非常有用。

表 1 IDAC8_CR0

位	7	6	5	4	3	2	1	0
值	保留			mode	Range[1:0]		hs	保留

- mode：将 DAC 设置为电压或电流模式。
- range[1:0]：DAC 范围设置。
- hs：用于设置数据速度。

表 2 IDAC8_CR1

位	7	6	5	4	3	2	1	0
值	保留		mx_data	reset_udb_en	mx_idir	idirbit	Mx_ioff	ioffbit

- mx_data：选择数据源。
- reset_udb_en：DAC 复位启用。
- mx_idir：DAC 电流方向控制的复用器选择。

- idirbit：DAC 电流方向的寄存器源。
- mx_off：DAC 电流关断控制的复用器选择。
- ioffbit：DAC 电流关断的寄存器源

表 3 IDAC8_DATA

位	7	6	5	4	3	2	1	0
值	Data[7:0]							

- Data[7:0]：DAC 数据寄存器。

直流和交流电气特性

除非另外指定，否则：典型值 = 25 °C，Vdda = 5.0 V，压差 = 1.0 V 最小值，这些规范适用于所有范围：0 至 31.875 µA、0 至 255 µA、0 至 2.04 mA。

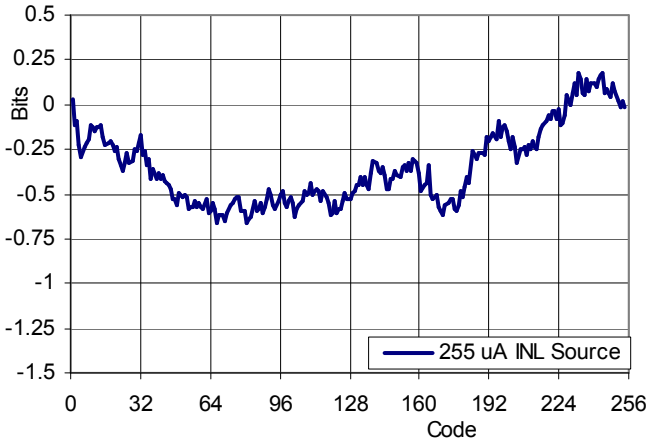
IDAC8 直流电特性

参数	说明	条件	最小值	典型值	最大值	单位
	分辨率		-	-	8	位
I_{OUT}	代码 = 255 处的输出电流	范围 = 2.048 mA，代码 = 255，V_{DDA} 2.7 V，Rload = 600	-	2.048	-	mA
		范围 = 2.048 mA，高速模式，代码 = 255，V_{DDA} 2.7 V，Rload = 300	-	2.048	-	mA
		范围 = 255 µA，代码 = 255，Rload = 600	-	255	-	µA
		范围 = 31.875 µA，代码 = 255，Rload = 600	-	31.875	-	µA
	单调性		-	-	是	
Ezs	零量程错误		-	0	±1	LSB
Eg	增益误差		-	-	3.5	%
INL	积分非线性	灌电流模式，范围 = 255 µA，代码 8 – 255，Rload = 2.4 k ，Cload = 15 pF	-	±1.2	±1.5	LSB
		电流源模式，范围 = 255 µA，代码 8 – 255，Rload = 2.4 k ，Cload = 15 pF	-	±0.9	±1	LSB

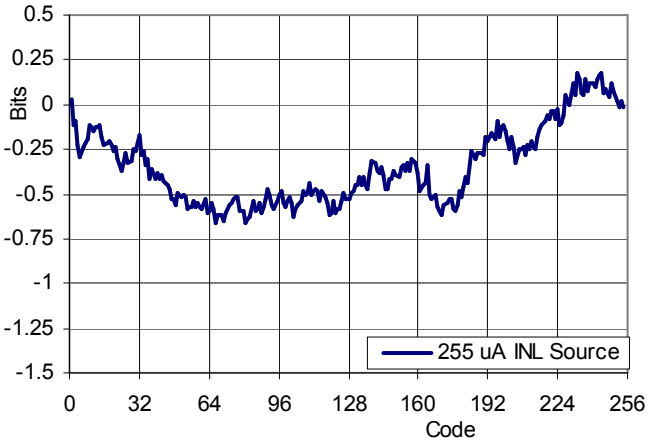
参数	说明	条件	最小值	典型值	最大值	单位
DNL	差分非线性	灌电流模式，范围 = 255 µA，Rload = 2.4 k ，Cload = 15 pF	–	±0.3	±0.5	LSB
		源电流模式，范围 = 255 µA，Rload = 2.4 k ，Cload = 15 pF	–	±0.3	±0.5	LSB
Vcompliance	压差电压、源电流或灌电流模式	电流最大时，Rload 与 Vdda，或 Rload 与 Vssa、Vdiff 与 Vdda 的压差	1	–	–	V
I_{DD}	工作电流，代码 = 255	慢速模式，源电流模式，范围 = 31.875 µA	–	–	44	µA
		慢速模式，源电流模式，范围 = 255 µA	–	–	33	µA
		慢速模式，源电流模式，范围 = 2.04 mA	–	–	33	µA
		慢速模式，灌电流模式，范围 = 31.875 µA	–	–	36	µA
		慢速模式，灌电流模式，范围 = 255 µA	–	–	33	µA
		慢速模式，灌电流模式，范围 = 2.04 mA	–	–	33	µA
		快速模式，源电流模式，范围 = 31.875 µA	–	–	310	µA
		快速模式，源电流模式，范围 = 255 µA	–	–	305	µA
		快速模式，源电流模式，范围 = 2.04 mA	–	–	305	µA
		快速模式，灌电流模式，范围 = 31.875 µA	–	–	310	µA
		快速模式，灌电流模式，范围 = 255 µA	–	–	300	µA
		快速模式，灌电流模式，范围 = 2.04 mA	–	–	300	µA

图形

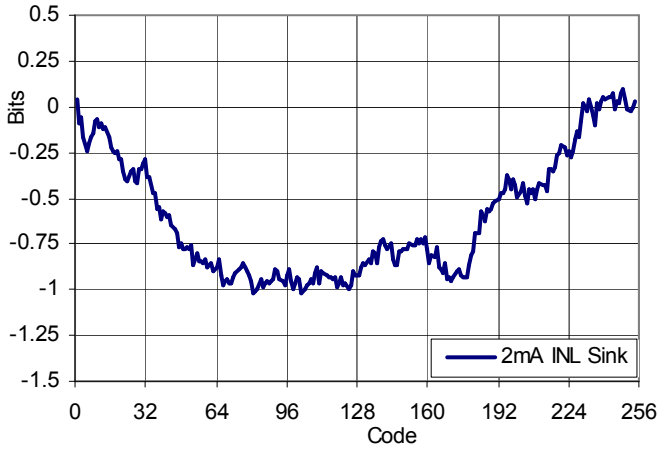
INL 与 DAC 代码，范围 = 255 µA，灌电流模式



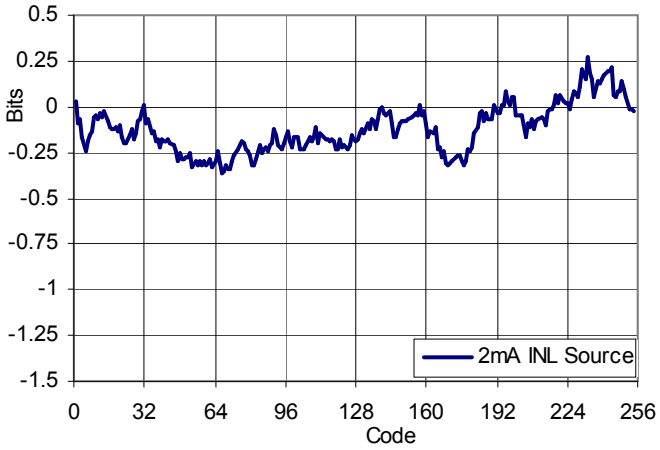
INL 与 DAC 代码，范围 = 255 µA，源电流模式



INL 与 DAC 代码，范围 = 2.04 mA，灌电流模式

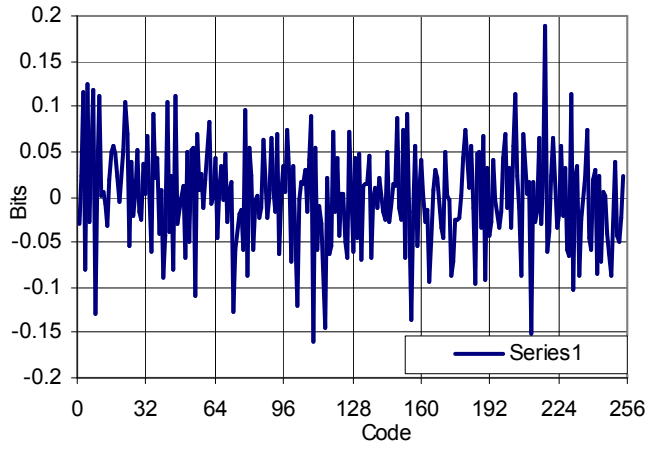


INL 与 DAC 代码，范围 = 2.04 mA，源电流模式

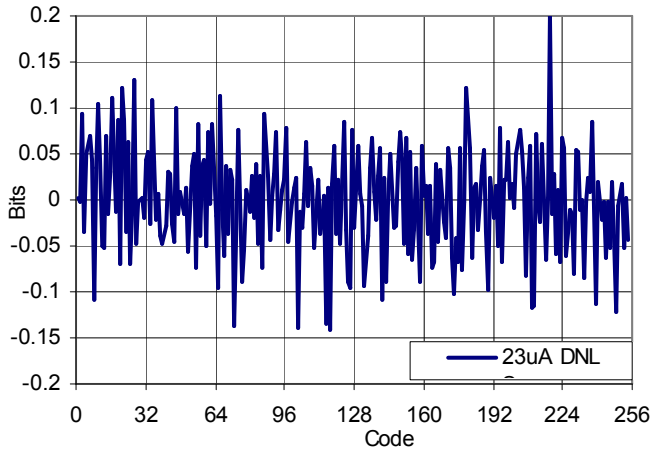


DNL 与 DAC 代码，范围 = 31.875 µA，灌电流模式

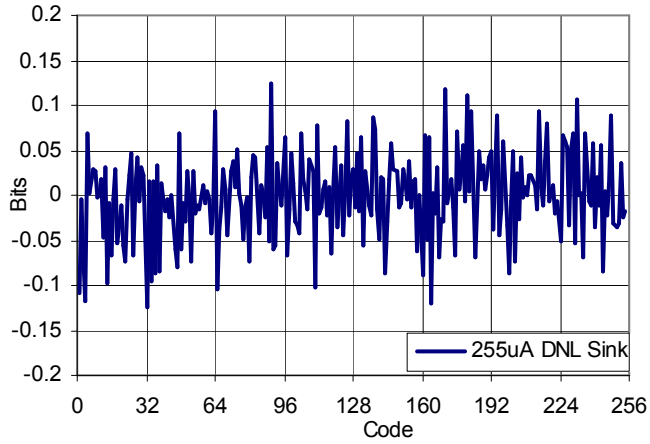
DNL 与 DAC 代码，范围 = 31.875 µA，源电流模式



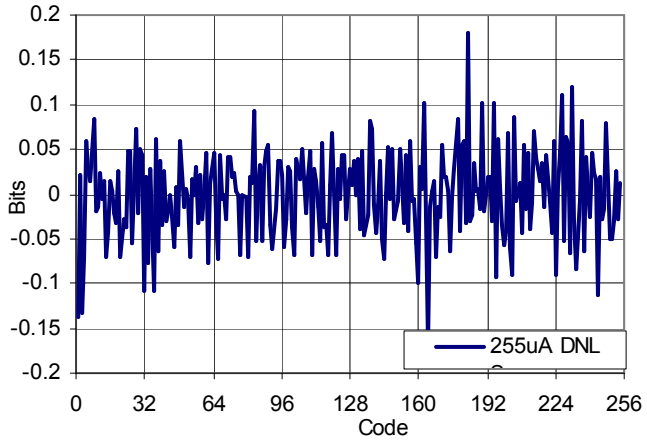
DNL 与 DAC 代码，范围 = 255 µA，灌电流模式



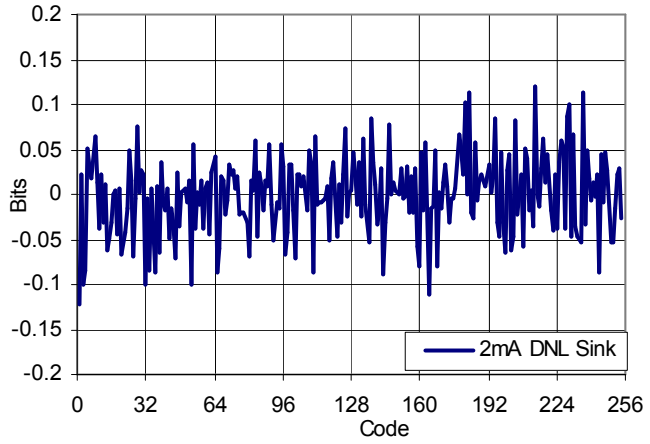
DNL 与 DAC 代码，范围 = 255 µA，源电流模式



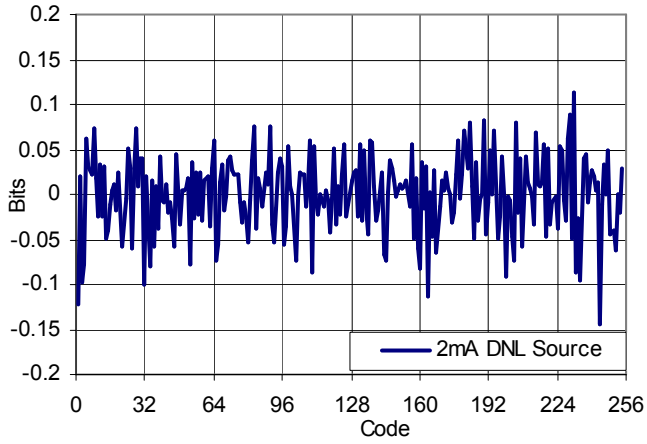
DNL 与 DAC 代码，范围 = 2.04 mA，灌电流模式



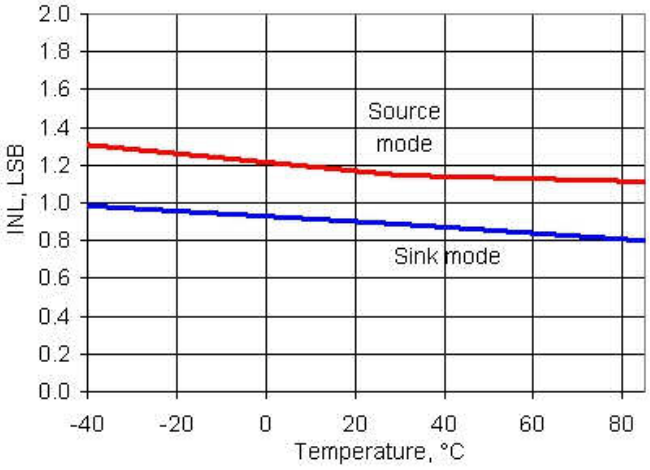
DNL 与 DAC 代码，范围 = 2.04 mA，源电流模式



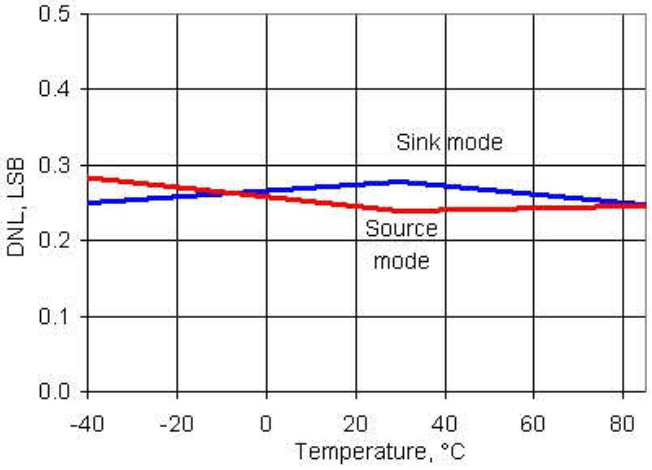
IDAC INL 与温度，范围 = 255 µA，快速模式



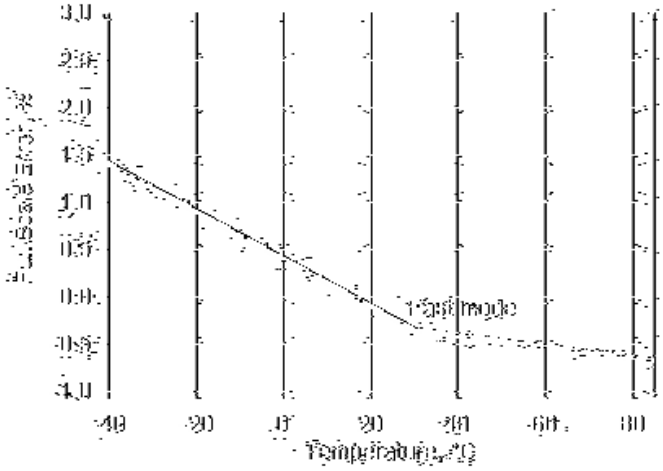
IDAC DNL 与温度，范围 = 255 µA，快速模式



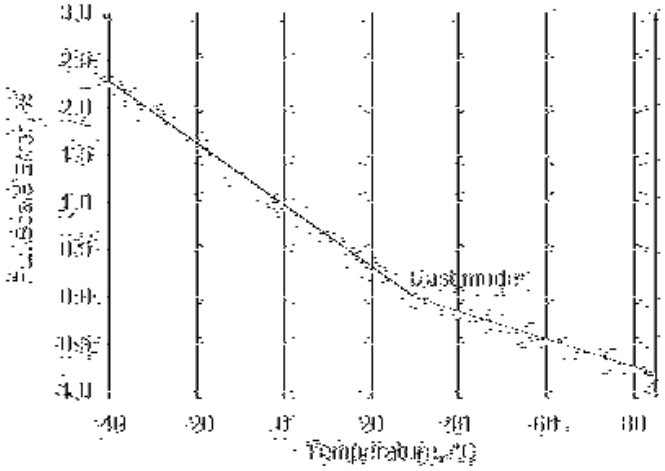
IDAC 全量程误差与温度，范围 = 255 µA，源电流模式



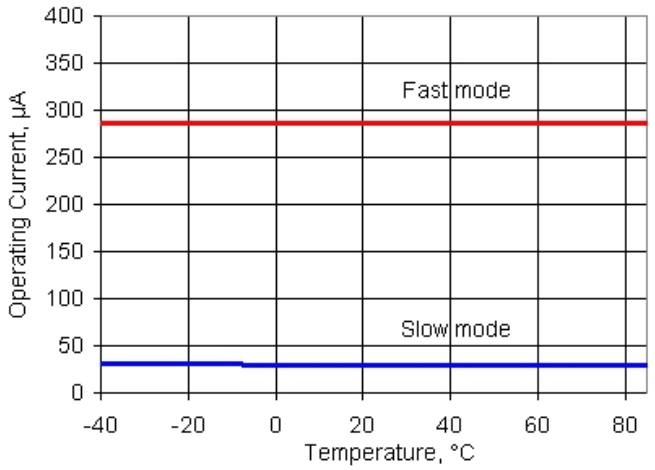
IDAC 全量程误差与温度，范围 = 255 µA，灌电流模式

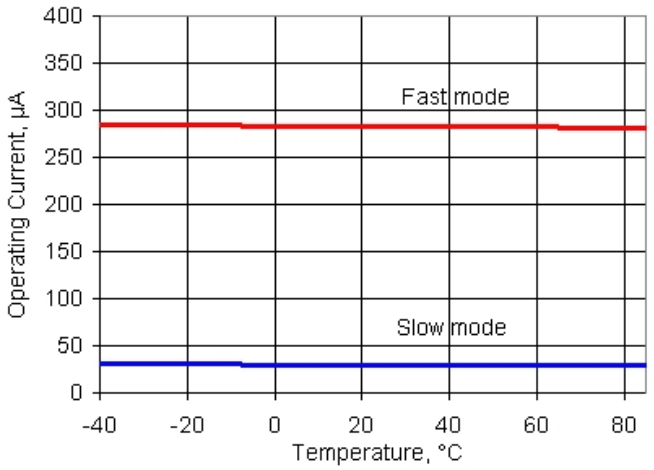


IDAC 工作电流与温度，范围 = 255 µA，代码 = 0，源电流模式



IDAC 工作电流与温度，范围 = 255 µA，代码 = 0，灌电流模式





IDAC8 交流电特性

参数	说明	条件	最小值	典型值	最大值	单位
F_{dac}	更新速率		–	–	8	Msps
T_{settle}	到 0.5 LSB 的建立时间	独立于 IDAC 范围设置 (I_{OUT})，全标度跃变，600 load，C_L = 15 pF，快速模式	–	–	100	ns
		独立于 IDAC 范围设置 (I_{OUT})，全标度跃变，600 load，C_L = 15 pF，慢速模式	–	–	1000	ns

组件更改

本节介绍组件与以前版本相比的主要更改。

版本	更改说明	更改/影响原因
1.60	添加了 GUI 配置编辑器	以前的配置窗口没有提供足够的易于使用的信息。
	向数据手册中添加了特性数据	
	对数据表进行了少量编辑和更新	
1.50	添加了睡眠/唤醒和初始化/启用 API。	为支持低功耗模式并提供常用接口，以单独控制大多数组件的初始化和启用。
	向组件中添加了 DMA 功能文件。	此文件允许 PSoC Creator 中的 DMA 向导工具支持 IDAC8。

© 赛普拉斯半导体公司，2011。此处所包含的信息可能会随时更改，恕不另行通知。除赛普拉斯产品的内嵌电路之外，赛普拉斯半导体公司不对任何其他电路的使用承担任何责任。也不根据专利或其他权利以明示或暗示的方式授予任何许可。除非与赛普拉斯签订明确的书面协议，否则赛普拉斯产品不保证能够用于或适用于医疗、生命支持、救生、关键控制或安全应用领域。此外，对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

PSoC® 是赛普拉斯半导体公司的注册商标，PSoC Creator™ 和 Programmable System-on-Chip™ 是赛普拉斯半导体公司的商标。此处引用的所有其他商标或注册商标归其各自所有者所有。

所有源代码（软件和/或固件）均归赛普拉斯半导体公司（赛普拉斯）所有，并受全球专利法规（美国和美国以外的专利法规）、美国版权法以及国际条约规定的保护和约束。赛普拉斯据此向获许可者授予适用于个人的、非独占性、不可转让的许可，用以复制、使用、修改、创建赛普拉斯源代码的派生作品、编译赛普拉斯源代码和派生作品，并且其目的只能是创建自定义软件和/或固件，以支持获许可者仅将其获得的产品依照适用协议规定的方式与赛普拉斯集成电路配合使用。除上述指定的用途之外，未经赛普拉斯的明确书面许可，不得对此类源代码进行任何复制、修改、转换、编译或演示。

免责声明：赛普拉斯不针对此材料提供任何类型的明示或暗示保证，包括（但不仅限于）针对特定用途的适销性和适用性的暗示保证。赛普拉斯保留在不做出通知的情况下对此处所述材料进行更改的权利。赛普拉斯不对此处所述之任何产品或电路的应用或使用承担任何责任。对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

产品使用可能受适用的赛普拉斯软件许可协议限制。