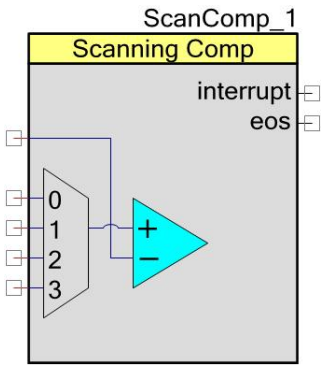
Scanning Comparator (ScanComp) 1.0



特性

- 自动扫描多达 64 个单端或各差分通道

  注意：被使用器件中可用的硬件限制输入和输出通道的数量。
- 64 个输出可路由至数字逻辑模块或引脚
- 多重比较模式

概述

扫描比较器（ScanComp）组件提供了硬件解决方案，以仅通过使用一个硬件比较器对 64 对模拟输入电压信号进行比较。可以使能样例比较器输出，用于数字硬件中的连接。可以将内部参考电压或外部电压连接到任一输入。

何时使用扫描比较器

通过扫描比较器组件可以对 64 个模拟信号进行比较。虽然 ADC 可以与软件配合使用来比较多个电压电平，但是扫描比较器更适用于需要快速响应或很少软件干预的应用场合。一些应用示例有：电源或模拟电平到数字信号的简单转换。当需要进行比较的信号多于器件中比较器的数量时，可以使用该组件。

输入/输出接口

本节介绍了扫描比较器的各种输入和输出接口。I/O 列表中的星号（*）表示 I/O 可能在某种 SAR ADC 配置下被隐藏。

时钟 — 数字输入*

比较器输入复用的时钟。使用内部时钟配置时，此时钟不存在。

vplus[n] — 模拟

差分对电压的正向输入，通常是进行比较的信号。

vminus[n] — 模拟*

差分对电压的负向输入，通常作为参考输入使用。当所有输入具有通用的负向参考输入或均使用了 VDAC 时，该负向输入不存在。

vminus — 模拟*

电压输入作为所有正向输入的负向参考输入。当所有输入具有通单独的参考输入，或使用 VDAC 时，此电压输入不存在。

中断 — 数字输出

所有通道上中断的“或运算”结果。

eos — 数字输出

扫描结束。

cmpout[n] — 数字输出*

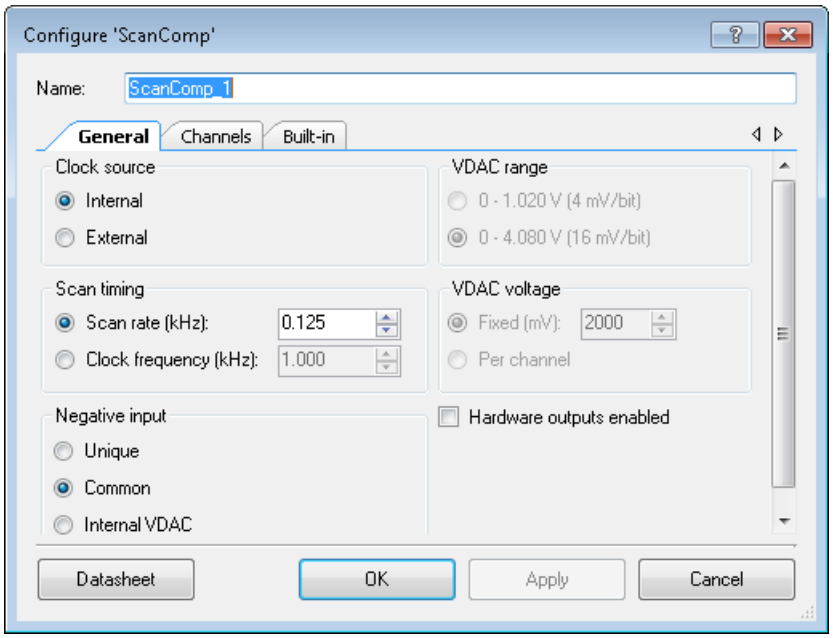
每个输入的单独比较器输出。

注意：这些输出在定制器中被禁用。

组件参数

将扫描比较器拖入设计中，双击它以打开 Configure（配置）对话框。该对话框包含了下列各选项卡和参数。

General 选项卡



Clock source（时钟源）

通过该参数来选择时钟实现序列：内部或外部。默认设置为内部。

Scan Rate（扫描速率）

当时钟源是内部时钟源时，此参数将扫描速率的单位设置为每秒周期数。此参数对时钟频率参数产生影响并受其影响。时钟源为外部时钟源时，此参数不可用。

扫描速率参数的有效范围取决于时钟频率参数的范围以及通道数量。

Clock Frequency（时钟频率）

时钟源为内部时钟源时，此参数将时钟频率的单位设置为 kHz。此参数对扫描速率参数产生影响并受其的影响。时钟源为外部时，频率选择不可用。在设计范围资源时钟编辑器中，实际的内部时钟频率显示为“ScanComp_Clock_int”。

Negative Input（负向输入）选项	时钟频率参数的范围（kHz）
Unique（独特）	1–10000。默认值为1。
Common（通用）（默认设置）	1–10000。默认值为1。
Internal VDAC（内部VDAC）。VDAC range（VDAC范围）：0–1.020 V	1–2000。默认值为1。
Internal VDAC（内部VDAC）。VDAC range（VDAC范围）：0–4.080 V	1–500。默认值为1。

Hardware Outputs Enabled（硬件输出使能）

该复选框用于使能此组件的锁存输出。默认情况下，该复选框被禁用。

Negative Input（负向输入）

此参数用于选择组件的负向输入模式。

Negative Input（负向输入）选项	说明
Unique（独特）	每个通道都有其单独的复用负向模拟输入。
Common（通用）（默认设置）	所有通道都使用了相同的负向模拟输入。
Internal VDAC（内部VDAC）	所有通道将组件配置的VDAC的输出用于它们的负载模拟输入。此模式不适用于PSoC 4器件。

VDAC Configuration - Range（VDAC 配置 — 范围）

通过该参数可以从两个电压范围中为 VDAC 选择一个。如果 Negative Input（负向输入）参数为 Unique（独特）或 Common（通用），则此参数不可用。

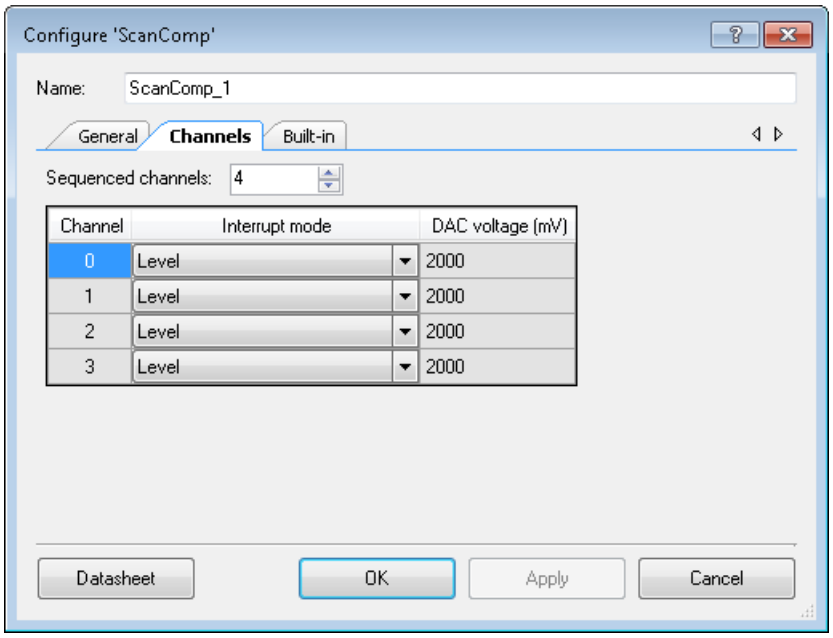
范围	最低值	最高值	步长	输出公式
0- 1.020 V	0.0 mV	1.020 V	4 mV	V_{OUT} = (value/256) × 1.024 V
0 – 4.080 V（默认）	0.0 mV	4.080 V	16 mV	V_{OUT} = (value/256) × 4.096 V

VDAC Configuration - Voltage（VDAC 配置 — 电压）

此参数用于选择内部 VDAC 的输出电压模式。如果 Negative Input（负向输入）参数为 Unique（独特）或 Common（通用）时，将不可用此参数。

电压	说明
固定（默认）	固定的初始值被写入到DAC内。在运行期间，通过调用API可以修改该值。
对于每个通道	每个通道具有其自己的DAC输出电压，该电压可以在Channels（通道）选项卡中配置。

Channels（通道）选项卡



Sequenced channels（已定序通道）

此参数可选择多达 64 个通道。被使用器件中可用的硬件限制输入和输出通道的数量。如果被使用的通道多于器件可支持的通道数量，该项目将无法构建。默认设置为 4。

Interrupt mode（中断模式）

此参数用于选择各通道的中断检测模式。

Interrupt mode（中断模式）	说明
Rising Edge（上升沿）	在输出上检测到上升沿时，将生成中断。
Falling Edge（下降沿）	输出上检测到下升沿时，将生成中断。

Interrupt mode（中断模式）	说明
Both Edges（上升沿和下降沿）	输出上检测到边沿（上升沿或下降沿）时，将生成中断。
Disable（禁用）	中断被禁用。
Level（电平）（默认）	输出为高电平时，将生成中断。

DAC Voltage （DAC 电压）

此参数用于设置各通道的初始电压值，单位为 mV。如果内部 VDAC 的输出电压模式为 Fixed（固定），则此参数不可用。默认设置为已选定的 DAC 范围（500 mV 或 2000 mV）的一半。

应用编程接口

通过应用编程接口（API），您可以使用软件进行配置组件。下面的表格列出并介绍了每个函数的接口。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“ScanComp_1”分配给提供的设计中的第一个组件实例。您可以将其重新命名为遵循标识符语法规则的任何唯一值。实例名称会成为与该组件相关的每个全局函数名称、变量和常量符号的前缀。为增加可读性，下表中使用的实例名称为“ScanComp”。

函数

函数	说明
ScanComp_Start()	对组件执行所有要求的初始化，并使能模块的电源。
ScanComp_Init()	根据定制器设置初始化或恢复组件。
ScanComp_Enable()	激活硬件并开始执行组件操作。
ScanComp_Stop()	关闭扫描比较器。
ScanComp_SetSpeed()	设置驱动功耗和速度。
ScanComp_SetDACRange()	将DAC设置为新范围。
ScanComp_GetDACRange()	获取DAC范围的设置情况
ScanComp_SetDACVoltage()	将DAC输出设置为新电压。
ScanComp_GetDACVoltage()	获取当前DAC输出电压的设置情况
ScanComp_SetChannelDACVoltage()	将特殊通道的DAC输出设置为新电压。
ScanComp_GetChannelDACVoltage()	获取特殊通道的DAC输出电压。

ScanComp_GetCompare()	获取已选通道的当前比较结果。
ScanComp_GetInterruptSource()	获取从已选模块发出的挂起中断请求。即使掩码中断也被返回
ScanComp_GetInterruptSourceMasked()	获取从已选模块发出的挂起中断请求。掩码中断不被返回。
ScanComp_GetInterruptMask()	获取已选模块中的当前中断掩码。
ScanComp_SetInterruptMask()	设置已选模块的中断屏蔽。
ScanComp_Sleep()	这是准备组件进入低功耗模式的首选API。
ScanComp_Wakeup()	此API函数是将组件恢复到调用ScanComp_Sleep() 前的状态。

全局变量

函数	说明
ScanComp_initVar	ScanComp_initVar变量用于说明此组件的初始配置。此变量前面添加了组件名称。该变量初始化为0，并在第一次调用ScanComp_Start()时设置为1。这允许进行组件初始化，而无需在所有对ScanComp_Start()例程的后续调用中再次初始化。 如果需要对该组件进行重新初始化，应在调用ScanComp_Init()和ScanComp_Enable()之前调用ScanComp_Stop()例程。

void ScanComp_Start(void)

说明： 对组件执行所有要求的初始化，并使能模块的电源。第一次执行子程序时，组件会初始化为定制器中的配置。根据配置采样率和比较器的响应时间规范设置功耗/速度。如果使用了外部时钟，该值将为最大值。在调用ScanComp_Stop()后，重启组件会保留当前组件的参数设置。

参数： 无

返回值： 无

副作用： 无

void ScanComp_Init(void)

说明： 根据定制器设置初始化或恢复组件。不需要调用ScanComp_Init()，因为ScanComp_Start() API会调用此函数，这是开始组件操作的优选方法。

参数： 无

返回值： 无

副作用： 无

void ScanComp_Enable(void)

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

参数： 无

返回值： 无

副作用： 无

void ScanComp_Stop(void)

说明： 通过关闭比较器、终止输入复用和关闭DAC（若使用）来关闭扫描比较器。

参数： 无

返回值： 无

副作用： 无

void ScanComp_SetSpeed(uint8 speed)

说明： 将驱动功耗和速度设置为以下三个设置之一：ScanComp_Start()根据配置采样率和比较器的响应时间规范设置功耗/速度。如果使用外部时钟，该值将为最大值。

参数： uint8 speed：枚举速度模式值

速度设置	注释
ScanComp_SLOWSPEED	慢速/超低功耗
ScanComp_MEDSPEED	中速/中等功耗
ScanComp_HIGHSPEED	高速/高功耗

返回值： 无

副作用： 无

void ScanComp_SetDACRange(uint8 DACRange)

说明： 将DAC设置为新范围。仅在选用了Internal DAC（内部DAC）时才使用此函数。

参数： uint8 DACRange — 写入到DAC内的新范围

范围	注释
ScanComp_INTERNAL_RANGE_1V	全量程范围为1.020 V
ScanComp_INTERNAL_RANGE_4V	全量程范围为4.080 V

返回值： 无

副作用： 无

uint8 ScanComp_GetDACRange(void)

说明： 获取DAC范围的设置。只在选用Internal DAC（内部DAC）的情况下才使用此函数。

参数： 无

返回值： uint8：代表DAC范围。

范围	注释
ScanComp_INTERNAL_RANGE_1V	全量程范围为1.020 V
ScanComp_INTERNAL_RANGE_4V	全量程范围为4.080 V

副作用： 无

void ScanComp_SetDACVoltage(uint8 DACVoltage)

说明： 将DAC输出设置为新电压。只在选用Internal DAC（内部DAC）的情况下才使用此函数。

参数： uint8 DACVoltage — 写入到DAC的电压，其单位取决于已选的电压范围（在1 V的范围内，单位为4 mV/位；在4 V的范围内，单位为16 mV/位）

返回值： 无

副作用： 无

uint8 ScanComp_GetDACVoltage(void)

说明： 获取当前DAC输出电压的设置。只在选用Internal DAC（内部DAC）的情况下才使用此函数。

参数： 无

返回值： uint8：将DAC配置为该电压，其单位取决于已选的范围（在1V范围内，单位为4 mV/位，在4 V范围内，单位为16 mV/位）

副作用： 无

void ScanComp_SetChannelDACVoltage(uint8 channel, uint8 DACVoltage)

说明： 将特殊通道的DAC输出设置为新电压。只在选用Internal DAC（内部DAC）而且电压为“Per channel”（每个通道）的情况下才使用此函数。

参数： uint8 channel — 指定需要修改DAC电压的通道。

uint8 DACVoltage — 将写入到DAC的电压，其单位取决于已选的范围（在1V范围内，单位为4 mV/位，在4 V范围内，单位为16 mV/位）

返回值： 无

副作用： 无

uint8 ScanComp_GetChannelDACVoltage(uint8 channel)

说明： 获取特殊通道的DAC输出电压。只在选用Internal DAC（内部DAC）而且电压为“Per channel”（每个通道）的情况下才使用此函数。

参数： uint8 channel — 指定需要返回DAC电压的通道。

返回值： uint8：将DAC配置为该电压，以用于给定的通道，其单位取决于已选的范围（在1 V范围内，单位为4 mV/位，在4 V范围内，单位为16 mV/位）

副作用： 无

uint8 ScanComp_GetCompare(uint8 channel)

说明： 获取已选通道的当前比较结果。

参数： uint8 channel — 指定需要返回比较结果的通道。

返回值： uint8：已选通道的比较结果。当输入值大于参考值时，将返回非零值。否则将返回0。

副作用： 无

uint8 ScanComp_GetInterruptSource(uint8 inputBlock)

说明： 获取从已选模块发出的挂起中断请求。此函数可确定生成中断的通道。即使掩码中断也被返回。此函数用于清除该输入模块的中断状态。

参数： uint8 inputBlock — 指定8个通道或更少的通道，它的中断请求将被返回。

返回值： uint8：已选定的8个输入或更少输入中的中断源位域。各输入的掩码值为：ScanComp_INTR_MASK。

副作用： 无

uint8 ScanComp_GetInterruptSourceMasked(uint8 inputBlock)

说明： 获取从已选模块发出的挂起中断请求。此函数可确定生成中断的通道。掩码中断不被返回。该函数清除中断状态。

参数： uint8 inputBlock — 指定8个通道或更少的通道，其的中断请求将被返回。

返回值： uint8：已选定的8个输入或更少输入中的中断源位域。各输入的掩码值为：ScanComp_INTR_MASK。

副作用： 无

uint8 ScanComp_GetInterruptMask(uint8 inputBlock)

说明： 获取已选模块中的当前中断掩码。该函数可确定当前掩码的通道中断。

参数： uint8 inputBlock — 指定8个通道或更少的通道，其的中断掩码将被返回。

返回值： uint8：已给通道数量的中断掩码。‘1’表示该中断被使能，‘0’表示该中断被掩码。各输入的掩码值为：ScanComp_INTR_MASK。

副作用： 无

void ScanComp_SetInterruptMask(uint8 inputBlock, uint8 mask)

说明： 为8个通道或更少的通道设置中断掩码。

参数： uint8 inputBlock — 指定8个通道或更少的通道，其的中断掩码将被写入。

uint8 mask — 指定的8个通道或更少通道的中断掩码值。‘1’表示使能了该中断，‘0’表示该中断被掩码。

返回值： 无

副作用： 无

void ScanComp_Sleep(void)

说明： 这是准备组件进入低功耗模式的首选API。扫描比较器不能在PSoC 3/5LP器件的睡眠模式和PSoC 4器件的深度睡眠模式下工作。

参数： 无

返回值： 无

副作用： 无

void ScanComp_Wakeup(void)

说明： 此API函数是将组件恢复到调用ScanComp_Sleep() 前的状态。

参数： 无

返回值： 无

副作用： 无

MISRA 合规性

本节介绍了 MISRA-C:2004 合规性和本组件的偏差情况。定义了两种类型的偏差：

- 项目偏差 — 适用于所有 PSoC Creator 组件的偏差
- 特定偏差 — 仅适用于本组件的偏差

本节提供了有关组件特定偏差的信息。*《系统参考指南》*的“MISRA 合规性”章节中介绍了项目偏差以及有关 MISRA 合规性验证环境的信息。

扫描比较器组件没有任何特定偏差。

该组件具有以下嵌入式组件：比较器、VDAC8、DMA、时钟、状态寄存器、控制寄存器。MISRA 合规性与特定偏差的相关信息，请参见相应的组件数据手册。

样例固件源代码

在 Find Example Project 对话框中，PSoC Creator 提供了大量的示例项目，包括原理图和示例代码。要获取组件示例，请打开组件目录中的对话框或原理图中的组件实例。要查看通用样例，请打开“Start Page”或 **File** 菜单中的对话框。根据要求，可以通过使用对话框中的 **Filter Options** 项来限定可选的项目列表。

有关更多信息，请参考 PSoC Creator 帮助中的 Find Example Project（查找示例项目）主题。

功能说明

扫描比较器组件仅允许通过使用一个硬件比较器便可以对多个信号集进行比较。需要在通过增加比较通道数量以缩短比较过渡的响应时间和使用其他硬件两种情况之间进行权衡。当需要进行比较的信号多于器件中比较器的数量时，可以使用该组件。

该组件使用硬件比较器（PSoC 4 器件中的 CTBm）和另一个硬件在输入和输出之间适当进行复用。通过使用模拟复用器，输入被自动复用为预定速率。一旦各输入稳定，比较器输出将被锁存，并且会根据配置中断模式生成中断输出。

进行比较后生成的中断被写入到状态寄存器内。使用所提供的 API 轮询状态寄存器时，可以在固件中观察单独比较器的中断状态。单独的组件范围中断输出信号是所有非掩码中断的“或”逻辑组合。

可以使能样例比较器输出，用于数字硬件中的连接。这样将需要使用另一个硬件，因此该方法是随意选择的。

扫描比较器组件在某个模式（属于三个比较模式中的一个）下工作。此模式通过定制器中的 Negative Input（负向输入）参数选择。

- Unique（独特） — 在该模式下，每个通道都有其自己的复用负向模拟输入。
- Common（通用） — 在该模式下，所有通道都使用相同的负向模拟输入。
- 内部 VDAC — 在该模式下，所有通道将组件配置的 VDAC 的输出用于它们的负载模拟输入。由于各通道通过 DMA 扫描，因此可以修改 VDAC 输出值，以适用于每个通道。此模式不适用于 PSoC 4 器件。

寄存器

扫描比较器组件具有一些状态寄存器。固件 API 使用这些寄存器来监控比较器输出值的状态和存储沿检测后生成的中断。用户固件不能直接访问其中任何寄存器。

资源

扫描比较器组件使用以下器件资源：

配置		资源							引脚数量（针对每个外部 I/O）
负向输入模式	已定序通道	宏单元	状态寄存器	控制寄存器	计数器7	比较器	DMA	VDAC	
独特	7	38	2	1	1	1	–	–	
	8	32	3						

配置		资源							引脚数量（针对每个外部I/O）
负向输入模式	已定序通道	宏单元	状态寄存器	控制寄存器	计数器7	比较器	DMA	VDAC	
	16	50	5						输入：1 [1] + 2 * Nch [2] 输出：2 + Nch [3]
	24	77	7						
	32	85	9						
通用	7	38	2	1	1	1	–	–	输入：1 + 1 [1] + Nch 输出：2 + Nch
	8	32	3						
	16	50	5						
	24	77	7						
	32	85	9						
内部VDAC/固定 [4]	7	38	2	1	1	1	–	1	输入：1 + Nch 输出：2 + Nch
	8	32	3						
	16	50	5						
	24	77	7						
	32	85	9						
内部VDAC/每个通道 [4]	7	39	2	1	1	1	1	1	输入：1 + Nch Outputs: 2+ Nch
	8	41	3						
	16	67	5						
	24	101	7						
	32	117	9						

API 存储器使用情况

根据不同编译器、组件、所使用的 API 数量以及组件的配置情况，组件所用的存储空间大小也不一样。下表提供了在某一组件配置中的所有 API 使用的存储器大小。

---

[1] 取决于时钟源参数。该输入不适用于内部时钟。

[2] Nch — 已定序通道的数量。

[3] 取决于硬件输出使能参数。

[4] 不适用于 PSoC 4 器件。

通过使用释放模式下相应的编译器，可以完成测量操作。在该模式下，存储器的大小得到优化。有关特定的设计，可分析编译器生成的映射文件以确定存储器使用情况。

配置		PSoC 3（Keil_PK51）		PSoC 4（GCC）		PSoC 5LP（GCC）	
负向输入模式	已定序通道	闪存（字节）	RAM（字节）	闪存（字节）	RAM（字节）	闪存（字节）	RAM（字节）
独特/通用	4	–	–	500	7	–	–
	8	698	5	–	–	688	5
	16	709	5	–	–	756	5
	24	724	5	–	–	780	5
	32	737	5	–	–	804	5
内部VDAC/固定	8	1055	10	–	–	1184	10
	16	1067	10	–	–	1248	10
	24	1080	10	–	–	1272	10
	32	1093	10	–	–	1300	10
内部VDAC/每个通道	8	1279	20	–	–	1328	20
	16	1292	28	–	–	1392	28
	24	1305	36	–	–	1416	36
	32	1318	44	–	–	1448	44

组件调试窗口

扫描比较器组件支持 PSoC Creator 组件调试窗口。有关每个寄存器的详细说明，请查看合适的器件数据手册。扫描比较器组件调试窗口中显示了以下寄存器。

寄存器	说明
ScanComp_Out1_8_STATUS_REG	1通道至8通道上的输出状态
ScanComp_Out9_16_STATUS_REG	9通道至16通道上的输出状态
ScanComp_Out17_24_STATUS_REG	17通道至24通道上的输出状态
ScanComp_Out25_32_STATUS_REG	25通道至32通道上的输出状态
ScanComp_Out33_40_STATUS_REG	33通道至40通道上的输出状态
ScanComp_Out41_48_STATUS_REG	41通道至48通道上的输出状态
ScanComp_Out49_56_STATUS_REG	49通道至56通道上的输出状态

寄存器	说明
ScanComp_Out57_64_STATUS_REG	57通道至64通道上的输出状态
ScanComp_Comp_Comp_PM_ACT_CFG	有效功耗模式配置寄存器7
ScanComp_Comp_Comp_PM_STBY_CFG	待机功耗模式配置寄存器7
ScanComp_Comp_Comp_TR0	比较器裁剪寄存器TR0
ScanComp_Comp_Comp_TR1	比较器裁剪寄存器TR1
ScanComp_Comp_Comp_CR	比较器控制寄存器
ScanComp_Comp_Comp_SW0	比较器模拟路由寄存器0
ScanComp_Comp_Comp_SW1	比较器模拟路由寄存器1
ScanComp_Comp_Comp_SW2	比较器模拟路由寄存器2
ScanComp_Comp_Comp_SW3	比较器模拟路由寄存器3
ScanComp_Comp_Comp_SW4	比较器模拟路由寄存器4
ScanComp_Comp_Comp_SW5	比较器模拟路由寄存器5
ScanComp_Comp_Comp_SW6	比较器模拟路由寄存器6
ScanComp_Comp_Comp_CLK	比较器时钟控制寄存器
ScanComp_Comp_Comp_WRK	比较器输出工作寄存器
VDAC8_PM_ACT_CFG	有效功耗模式配置寄存器8
VDAC8_PM_STBY_CFG	待机功耗模式配置寄存器7
VDAC8_TR	DAC裁剪寄存器
VDAC8_CR0	DAC模块控制寄存器0
VDAC8_CR1	DAC模块控制寄存器1
VDAC8_SW0	DAC模拟路由寄存器0
VDAC8_SW2	DAC模拟路由寄存器1
VDAC8_SW3	DAC模拟路由寄存器2
VDAC8_SW4	DAC模拟路由寄存器3
VDAC8_STROBE	DAC探针寄存器
VDAC8_DATA	DAC数据寄存器

直流和交流的电气特性

下面各值表示其预计性能，它们基于初始特性数据。

扫描比较器直流规范

参数	说明	条件	最小值	典型值	最大值	单位
V_{OS}	高速/高功耗模式 [5]中的输入偏移电压	出厂预设值，V_{DDA} > 2.7 V，V_{IN} ≥ 0.5 V	–	–	10	mV
	中速/中等功耗模式 [5]中的输入偏移电压	出厂预设值，V_{IN} ≥ 0.5 V	–	–	9	mV
	慢速/超低功耗模式 [5]中的输入偏移电压	V_{DDA} ≤ 4.6 V	–	±12	–	mV
Icmp	静态电流		–	–	Icmp = Idac [6] + Icomp [7] + Ibase [8]	uA
CMRR	共模抑制比		30	50		dB
V_{CMP}	输入共模电压 [5]	超低功耗模式	0		V_{DDA} – 1.1	V
		中等功耗模式	0		V_{DDA} – 0.1	V
		高功耗模式	0		V_{DDA} – 0.1	V

扫描比较器交流规范

参数	说明	条件	最小值	典型值	最大值	单位
fclk	时钟频率	没有内部VDAC	0.001	–	10	MHz
		有内部VDAC（量程 – 1V）	0.001	–	2	MHz
		有内部VDAC（量程 – 4V）	0.001	–	0.5	MHz

5 可以在 Hover 窗口中的组件上显示观察速度功耗参数（该参数取决于内部时钟或外部时钟的高功耗）。

6 Idac — VDAC 工作电流。在 VDAC8 组件数据手册中介绍了该值。组件的负向输入模式为 Unique（独特）或 Common（通用）时，将不使用 VDAC。

7 Icomp — 比较器的工作电流。该值在比较器组件数据手册中介绍。

8 Ibase — UDB 模块的电流。在扫描比较器组件中，该电流的典型值为 45 uA / MHz。

参数	说明	条件	最小值	典型值	最大值	单位
T_{RESP}	响应时间[5]	超低功耗模式	55 / 1 [9]	–	(2*Nch)/fclk [10]	us
		中等功耗模式	0.155 / 0.4 [9]	–	(2*Nch)/fclk [10]	us
		高功耗模式	0.075 / 0.15 [9]	–	(2*Nch)/fclk [10]	us

组件更改

本节列出了组件与先前版本相比的主要更改的内容。

版本	更改说明	更改原因/影响
1.0	新组件。	

赛普拉斯半导体公司，2013-2016 年。本文件是赛普拉斯半导体公司及其子公司，包括 Spansion LLC（“赛普拉斯”）的财产。本文件，包括其包含或引用的任何软件或固件（“软件”），根据全球范围内的知识产权法律以及美国与其他国家签署条约由赛普拉斯所有。除非在本款中另有明确规定，赛普拉斯保留在该等法律和条约下的所有权利，且未就其专利、版权、商标或其他知识产权授予任何许可。如果软件并不附随有一份许可协议且贵方未以其他方式与赛普拉斯签署关于使用软件的书面协议，赛普拉斯特此授予贵方属人性质的、非独家且不可转让的如下许可（无再许可权）（1）在赛普拉斯特软件著作权项下的下列许可权（一）对以源代码形式提供的软件，仅出于在赛普拉斯硬件产品上使用之目的且仅在贵方集团内部修改和复制软件，和（二）仅限于在有关赛普拉斯硬件产品上使用之目的将软件以二进制代码形式的向外部最终用户提供（无论直接提供或通过经销商和分销商间接提供），和（2）在被软件（由赛普拉斯公司提供，且未经修改）侵犯的赛普拉斯专利的权利主张项下，仅出于在赛普拉斯硬件产品上使用之目的制造、使用、提供和进口软件的许可。禁止对软件的任何其他使用、复制、修改、翻译或汇编。

在适用法律允许的限度内，赛普拉斯未对本文件或任何软件作出任何明示或暗示的担保，包括但不限于关于适销性和特定用途的默示保证。赛普拉斯保留更改本文件的权利，届时将不另行通知。在适用法律允许的限度内，赛普拉斯不对因应用或使用本文件所述任何产品或电路引起的任何后果负责。本文件，包括任何样本设计信息或程序代码信息，仅为供参考之目的提供。文件使用人应负责正确设计、计划和测试信息应用和由此生产的任何产品的功能和安全性。赛普拉斯产品不应被设计为、设定为或授权用作武器操作、武器系统、核设施、生命支持设备或系统、其他医疗设备或系统（包括急救设备和手术植入物）、污染控制或有害物质管理系统中的关键部件，或产品植入之设备或系统故障可能导致人身伤害、死亡或财产损失其他用途（“非预期用途”）。关键部件指，若该部件发生故障，经合理预期会导致设备或系统故障或会影响设备或系统安全性和有效性的部件。针对由赛普拉斯产品非预期用途产生或相关的任何主张、费用、损失和其他责任，赛普拉斯不承担全部或部分责任且贵方不应追究赛普拉斯之责任。贵方应赔偿赛普拉斯因赛普拉斯产品任何非预期用途产生或相关的所有索赔、费用、损失和其他责任，包括因人身伤害或死亡引起的主张，并使之免受损失。

赛普拉斯、赛普拉斯徽标、Spansion、Spansion 徽标，及上述项目的组合，WICED，及 PSoC、CapSense、EZ-USB、F-RAM 和 Traveo 应视为赛普拉斯在美国和其他国家的商标或注册商标。请访问 cypress.com 获取赛普拉斯商标的完整列表。其他名称和品牌可能由其各自所有者主张为该方财产。

[9] PSoC 4 器件的响应时间。

[10] Nch — 已定序通道的数量；fclk = 时钟频率。