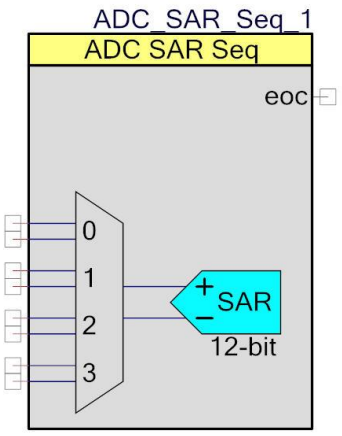
序列逐次逼近ADC（ADC_SAR_Seq）

1.0

特性



- 支持 PSoC 5LP 器件
- 可选分辨率（8、10 或 12 位）和采样率

（高达 1 Msps）

- 自动扫描多达 64 个单端或 32 个差分通道

注意：输入通道的实际最大数量取决于特定 PSoC 器件和封装中可路由模拟 GPIO 的数量。

概述

通过序列 SAR ADC 组件，用户能够在 PSoC 5LP 上轻易配置和使用不同操作模式的 SAR ADC。它提供原理图级别和固件级别的支持，可在 PSoC Creator 的设计和工程中方便的使用序列 SAR ADC。用户可以配置多个模拟通道被自动扫描，其扫描结果被保存在单独的 SRAM 位置。

使用 ADC_SAR_Seq 的情况

在采样率高的系统（必须采样多个通道）中，最常使用的组件是序列 SAR ADC。使用硬件来完成整个序列，这样能够降低高速率系统的 CPU 负载。该组件以 ADC 硬件所支持的最快速度对各个通道进行排序，以使可能的采样率最大。

输入/输出接口

本节介绍 ADC_SAR_Seq 的输入和输出接口。I/O 列表中的星号(*)表示，在 I/O 说明部分中所列出的特定条件下，该 I/O 可能不可见。

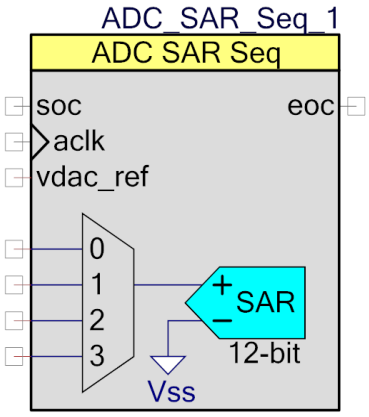


图 1. 单端输入的 ADC_SAR_Seq

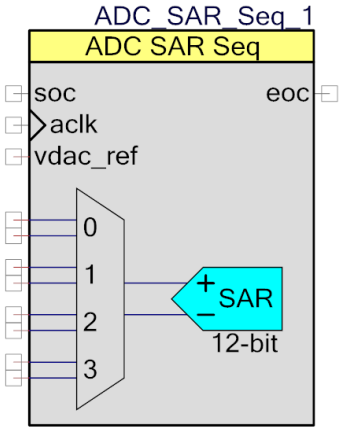


图 2. 差分输入的 ADC_SAR_Seq（每对中，高的输入作为正向输入，低的作为负向输入）

+输入 — 模拟

此输入是 ADC_SAR_Seq 的正向模拟信号输入。转换结果是一个+输入信号减去电压参考的函数。电压参考是 –输入信号或 V_{SSA}。在差分配置中，即为每个通道的高输入。

–输入 — 模拟*

显示时，此可选输入是 ADC_SAR_Seq 的负向模拟信号（或参考）输入。转换结果取决于+输入减去–输入。在将 **Input Range**（输入范围）参数设置为差分模式之一时可看到此引脚。

vdac_ref — 输入*

VDAC 参考（vdac_ref）是可选引脚。只有将输入范围设置为 **Vssa 至 VDAC*2（单端模式）**或 **0.0 +/- VDAC（差分模式）**时，此引脚才可见。

注意：只能将此引脚连接到 VDAC 组件输出。请勿将其连接到任何其他信号。

soc — 输入*

开始转换（soc）是可选引脚。仅在 **Triggered**（触发）采样模式中才可见到此引脚。该输入的上升沿将启动所有通道进行扫描。此信号必须同 ADC_SAR_Seq 时钟同步。将 **Sample Mode**（采样模式）参数设置为 **Free Running**（持续运行）时，此引脚不可见。

注意：如果此输入被连接到一个恒定的逻辑高电平，则组件的运行情况与持续运行模式中的相同，只是不能调用 StartConvert() API。

aclk — 输入 *

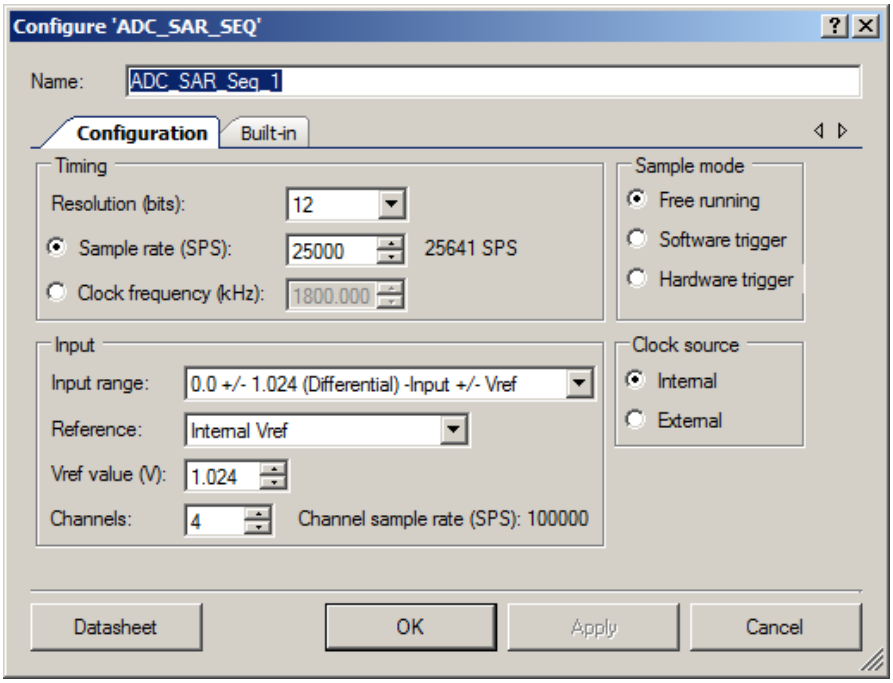
如果 **Clock Source**（时钟源）参数设置为 **External**（外部），则显示该引脚。该时钟确定了作为转换方法和分辨率函数的转换速率。

eoc — 输出

此信号的上升沿表示已经完成了所有通道的扫描过程，并且扫描结果被传输到 SRAM 阵列。即使您也可以连接一个备用的中断组件，仍将一个组件内部中断连接到此信号。

组件参数

将一个 ADC_SAR_Seq 组件拖放到您的设计窗口上，并双击以打开 **Configure**（配置）对话框。



ADC_SAR_Seq 具有以下参数。粗体表示默认选项。

模式

Resolution（分辨率）

设置 ADC_SAR_Seq 的分辨率。

分辨率	说明
12	将分辨率设置为12位
10	将分辨率设置为10位
8	将分辨率设置为8位

Conversion Rate（转换速率）

此参数用于设置 ADC 的转换速率，单位为每秒采样数“SPS”。转换时间与转换速率成反比。以每秒采样数为单位输入转换速率。转换一个采样需要 18 个组件时钟周期。

Clock Frequency（时钟频率）

该文本框为只读（始终为灰色）区域，用于显示所选工作条件（分辨率和转换速率）所需的时钟速率。当这两个条件中的任一条件更改或两个条件都更改时，会更新该文本框。时钟频率可以为介于 1 MHz 与 18 MHz 之间的任何值。占空比应为 50%。最小脉冲宽度应大于 25.5 ns。如果时钟不在这些限制范围内，则 PSoC Creator 会在编译过程中生成错误。在这种情况下，在设计范围资源时钟编辑器中更改主控时钟。

Sample Mode（采样模式）

该参数确定了 ADC 的工作方式。

采样模式	说明
自由运行	ADC_SAR_Seq持续运行
软件触发器	StartConvert() API的调用会启动所有通道的一次转换
硬件触发	SOC引脚上的一个上升沿脉冲触发一次转换。

时钟源

通过此参数可以选择 ADC_SAR_Seq 组件的内部时钟或外部时钟。

ADC时钟	说明
内部	使用ADC_SAR_Seq的内部时钟。
外部	使用外部时钟。时钟源可以是模拟、数字或由其他组件生成。

Input（输入）

Input Range（输入范围）

该参数按给定的输入范围配置 ADC。无论使用了什么输入范围设置，连接至 PSoC 的模拟信号都必须在 V_{SSA} 与 V_{DDA} 之间。

单端模式输入范围	说明
0.0至2.048V **0至Vref*2**	当使用内部参考电压（1.024 V）时，可用输入范围为 0.0至2.048 V。对ADC进行配置，以便使用内部连接到Vrefhi_out的负向输入。如果使用外部参考电压，可用的输入范围为0.0至Vref*2。
Vssa至Vdda	该模式使用V_{DDA}/2参考电压。可用输入范围涵盖完整模拟电源电压。ADC置于单端输入模式，同时–输入在内部连接到Vrefhi_out。如果使用外部参考电压，可用的输入范围为0.0至Vref*2。

单端模式输入范围	说明
Vssa至VDAC*2	此模式使用VDAC参考，该参考应连接到vdac_ref引脚。可用的输入范围为Vssa到VDAC*2伏。对ADC进行配置，以便通过内部连接到Vrefhi_out的负向输入。

差分输入范围	说明
0.0 ± 1.024V –输入 ± Vref	当使用内部参考（1.024 V）时，输入范围为–输入 ± 1.024 V。 例如，如果–输入连接到2.048 V，则可用输入范围为2.048 ± 1.024 V或1.024至3.072 V。对于需要扫描单端和差分信号的系统，在扫描单端输入时将–输入连接到Vssa。 您可以使用外部参考源提供更宽的操作范围。可以使用相同等式“–输入 ± Vref”来计算可用的输入范围。
0.0 ± Vdda –输入 ± Vdda	该模式与供电电压成比例。输入范围为–输入± Vdda。对于需要扫描单端和差分信号的系统，在扫描单端输入时，将负向输入连接至Vssa。如果使用外部参考电压，则可用输入范围为–输入 ± Vref。
0.0 ± Vdda/2 –输入 ± Vdda/2	该模式与供电电压成比例。输入范围为–输入 ± Vdda/2。对于需要扫描单端和差分信号的系统，在扫描单端输入时，将负向输入连接至Vssa。如果使用外部参考电压，则可用输入范围为–输入 ± Vref。
0.0 ± VDAC –输入 ± VDAC	此模式使用VDAC参考，该参考应连接到vdac_ref引脚。输入范围为–输入 ±VDAC。对于需要扫描单端和差分信号的系统，在扫描单端输入时，将负向输入连接至Vssa。

Reference（参考电压）

此参数用于选择 ADC_SAR 的参考电压配置。

参考电压	说明
Internal Vref（内部Vref）	使用内部参考。该选项允许的最大采样率为100,000 sps。对于较高采样率，请使用**Internal Vref, bypassed**选项。
Internal Vref, bypassed（内部Vref，旁路）	使用内部参考和SAR1的引脚P0[2]*上或SAR0的引脚P0[4]*上的外部旁路电容器。
External Vref（外部Vref）	在SAR1的引脚P0[2]*上或SAR0的引脚P0[4]上使用外部参考。

***** 如果内部数字开关引发的噪音超过某个应用对模拟性能的要求，建议使用外部旁路电容。若要使用该选项，请将端口引脚 P0[2]或 P0[4]配置为模拟 HI-Z 引脚，并连接到 0.01 µF 至 10 µF 范围内的外部电容。

注意：ADC_SAR 和 ADC_DelSig 组件使用相同的内部参考。如果两类 ADC 需要同时使用内部参考，即使 SAR ADC 采样率低于 100 ksps，也要使用 **Internal Vref, bypassed** 选项以获得最佳性能。

Voltage Reference（电压参考）

电压参考用于计入应用编程接口一节中讨论的电压转换函数的 ADC。当使用内部参考时，该参数读。当使用外部参考时，可以编辑该值以匹配外部参考电压。

- 选择输入范围 **Vssa 至 Vdda**、**-输入 +/- Vdda** 或**-输入 +/- Vdda/2** 时，其值由“设计范围资源”（DWR）窗口中系统选项卡内的 V_{DDA} 设置演变而来。
- 当选择输入范围 **Vssa 至 VDAC*2** 或**–输入 +/- VDAC** 时，请输入 VDAC 电源电压值。

注意：输入范围和参考电压的限制为 Vssa 至 Vdda 电压。

应用编程接口

通过应用程序编程接口（API）例程，为 ADC_SAR_Seq 组件提供了各控制和配置函数。下面的表格列出并说明了每个函数的接口。后面部分将更详细地介绍每个函数。

默认情况下，PSoC Creator 将实例名称“ADC_SAR_Seq _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。实例名称会成为所有全局函数名称、变量和符号常量的前缀。出于可读性考虑，下表中使用的实例名称为“ADC_SAR_Seq”。

函数	说明
ADC_SAR_Seq_Start()	给ADC_SAR_Seq加电并复位所有状态
ADC_SAR_Seq_Stop()	停止ADC_SAR_Seq转换并将功耗减少到最小值
ADC_SAR_Seq_SetResolution()	设置分辨率
ADC_SAR_Seq_StartConvert()	开始转换
ADC_SAR_Seq_StopConvert()	停止转换
ADC_SAR_Seq_IRQ_Enable()	内部IRQ连接到“eoc”信号。该API启用了内部ISR。
ADC_SAR_Seq_IRQ_Disable()	内部IRQ连接到“eoc”信号。该API禁用了内部ISR。
ADC_SAR_Seq_IsEndConversion()	如果转换完成，将返回非零值
ADC_SAR_Seq_GetAdcResult()	返回带符号的10位转换结果，该结果被存储在ADC SAR数据寄存器内，而不是存储在结果缓冲区内
ADC_SAR_Seq_GetResult16()	返回特定通道上带符号的16位转换结果
ADC_SAR_Seq_SetOffset()	设置偏移量
ADC_SAR_Seq_SetGain()	设置增益，其单位为每伏计数
ADC_SAR_Seq_CountsTo_Volts()	将计数转换为单位为伏的浮点电压值

函数	说明
ADC_SAR_Seq_CountsTo_mVolts()	将计数转换为单位为毫伏的电压值
ADC_SAR_Seq_CountsTo_uVolts()	将计数转换为单位为微伏的电压值
ADC_SAR_Seq_Sleep()	停止ADC_SAR_Seq操作，并保存用户配置
ADC_SAR_Seq_Wakeup()	恢复并使能用户配置
ADC_SAR_Seq_Init()	初始化自定义程序所提供的默认配置
ADC_SAR_Seq_Enable()	启用ADC_SAR_Seq的时钟，并使其通电
ADC_SAR_Seq_Disable()	禁用组件，但不禁用ADC SAR
ADC_SAR_Seq_SaveConfig()	保存当前用户配置
ADC_SAR_Seq_RestoreConfig()	恢复用户配置

全局变量

变量	说明
ADC_SAR_Seq_initVar	该变量指示ADC是否已初始化。该变量初始化为0，并在第一次调用ADC_SAR_Seq _Start()时设置为1。这样，第一次调用ADC_SAR_Seq _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ADC_SAR_Seq _Start()或ADC_SAR_Seq _Enable()函数前调用ADC_SAR_Seq _Init()函数。
ADC_SAR_Seq_finalArray	此阵列包含在生成每个EOC脉冲后所有通道的有效转换结果内。 **注意：**如果使用该变量直接访问转换结果而没有使用GetResult16() API，那么需要遵循结果的前后顺序。在此阵列中，各个结果按照倒序放置，所以最后一个通道的结果被放置在ADC_SAR_Seq_finalArray[0]处。组件配置为差分输入模式时，不能直接读取转换结果，因为还没有补偿ADC SAR偏移量值给这些结果。使用GetResult16() API来返回转换结果时，GetResult16() API会进行补偿。

void ADC_SAR_Seq_Start(void)

说明： 此函数用于配置ADC_SAR_Seq，并为组件加电。调用此函数后，通过调用StartConvert函数或通过检测“eoc”信号的上升沿（若选中），可以启动ADC。这是开始执行组件操作的首选方法。ADC_SAR_Seq_Start()设置initVar 变量，调用ADC_SAR_Seq_Init()函数，然后调用ADC_SAR_Seq_Enable()函数。

参数： 无

返回值： 无

其他影响： 如果已设置initVar变量，则该函数仅调用ADC_SAR_Seq_Enable()函数。

void ADC_SAR_Seq_Stop(void)

说明： 停止ADC_SAR_Seq转换并将功耗减少到最小值。

注意：该API不会断开ADC电源，但是会将电源降到最小值。该器件有一个缺陷，即：器件断电时，与某些模拟资源的连接会不可靠。当停止使用该资源的组件时，这种不可靠性会在静默失败中表现出来（例如模拟组件中出现不可预见的失败结果）。

参数： 无

返回值： 无

其他影响： 无

void ADC_SAR_Seq_SetResolution(uint8 resolution)

说明： 设置GetResult16()和GetAdcResult() API的分辨率。

参数： uint8 resolution：设置分辨率

参数名称	值	说明
ADC_SAR_Seq_BITS_12	12	将分辨率设置为12位。
ADC_SAR_Seq_BITS_10	10	将分辨率设置为10位。
ADC_SAR_Seq_BITS_8	8	将分辨率设置为8位。

返回值： 无

其他影响： 在转换周期过程中不能更改ADC分辨率。建议的最佳实践是使用ADC_SAR_Seq_StopConvert()停止转换，更改分辨率，然后使用ADC_SAR_Seq_StartConvert()重新启动转换。

如果决定在调用此 API 之前不停止转换，请使用ADC_SAR_Seq_IsEndConversion()等到转换完成，然后再更改分辨率。

如果在转换过程中调用ADC_SAR_Seq_SetResolution()，则在当前转换完成之前，分辨率不会更改。对于新的分辨率，在另外的6 + “新分辨率（以位为单位）”个时钟周期内，不会以新的分辨率查看数据。

通过在ADC计数和施加的输入电压之间计算正确转换，可影响ADC_SAR_Seq_CountsTo_Volts()、ADC_SAR_Seq_CountsTo_mVolts()和ADC_SAR_Seq_CountsTo_uVolts()。计算取决于分辨率、输入范围和电压参考。

void ADC_SAR_Seq_StartConvert(void)

说明： 强制ADC启动一个转换。在自由运行模式中，ADC_SAR_Seq连续运行。在软件触发模式中，该函数还充当软件版本的SOC，并且必须通过ADC_SAR_Seq_StartConvert()触发每个转换。

在硬件触发模式中，该API无效。

参数： 无

返回值： 无

其他影响： 无

void ADC_SAR_Seq_StopConvert(void)

说明： 在自由运行模式中，强制ADC_SAR_Seq停止转换。如果当前正在执行转换，将会完成该转换，并且没有启动其他转换。

参数： 无

返回值： 无

其他影响： 无

void ADC_SAR_Seq_IRQ_Enable(void)

说明： 转换结束后，启用中断。必须启动全局中断，以实现ADC中断。要启用全局中断，请在*main.c*文件中调用启用全局中断宏“CyGlobalIntEnable;”。

参数： 无

返回值： 无

其他影响： 启用中断。读取结果将清除中断。

void ADC_SAR_Seq_IRQ_Disable(void)

说明： 转换结束后，禁用中断。

参数： 无

返回值： 无

其他影响： 无

uint8 ADC_SAR_Seq_IsEndConversion(uint8 retMode)

说明： 根据retMode参数的设置情况，会立即返回转换状态或在转换完成前（阻止）不返回转换状态。

参数： uint8 retMode：检查转换返回模式。有关各选项，请参见下表。

选项	说明
ADC_SAR_Seq_RETURN_STATUS	立即返回状态。如果返回值为零，则表示转换未完成，应重试该函数，直至返回非零值。
ADC_SAR_Seq_WAIT_FOR_RESULT	在ADC_SAR_Seq转换完成之前不返回结果。

返回值： uint8：如果返回非零值，则最后一次转换已完成。如果返回值为零，则ADC_SAR_Seq仍在计算最后的结果。

其他影响： 该函数可读取转换结束的状态，并在读取后会清除。

int16 ADC_SAR_Seq_GetAdcResult(void)

说明： 直接从ADC的数据寄存器内返回转换结果，而不是从结果缓冲区内。一般情况下会使用GetResult16()函数。ADC计数值被转换为2进制补码格式。

参数： 无

返回值： int16：最后ADC转换的16位结果。

其他影响： 无

int16 ADC_SAR_Seq_GetResult16(uint16 chan)

说明： 返回通道“chan”的转换结果，该结果来自结果缓冲区。计数值被转换为2进制补码格式。

参数： uint16 chan：通道编号

返回值： int16：返回转换数据如有符号的16位整数电压值

其他影响： 无

void ADC_SAR_Seq_SetOffset(int32 offset)

说明： 设置ADC_SAR_Seq_CountsTo_Volts()、ADC_SAR_Seq_CountsTo_mVolts()和ADC_SAR_Seq_CountsTo_uVolts()所用的ADC_SAR_Seq偏移，以便在计算电压转换前，从给定读数中减去该偏移量。

参数： int32 offset：如果该输入短路或连接到相同输入电压，则该值为测量值。

返回值： 无

其他影响： 通过减去给定偏移，可影响ADC_SAR_Seq_CountsTo_Volts()、ADC_SAR_Seq_CountsTo_mVolts()和ADC_SAR_Seq_CountsTo_uVolts()。

void ADC_SAR_Seq_SetGain(int32 adcGain)

说明： 为以下电压转换函数设置每伏电压的ADC_SAR_Seq增益计数。默认情况下，该值由参考和输入范围设置设定。该值仅可用于进一步校准具有已知输入的ADC_SAR_Seq，或仅在ADC_SAR_Seq使用外部参考的情况下使用。

参数： int32 adcGain：每伏电压的ADC_SAR_Seq增益计数。

返回值： 无

其他影响： 通过在ADC计数和采用的输入电压之间进行正确转换，可影响ADC_CountsTo_Volts()、ADC_CountsTo_mVolts()和ADC_CountsTo_uVolts()函数。

float32 ADC_SAR_Seq_CountsTo_Volts(int16 adcCounts)

说明： 将ADC_SAR_Seq输出转换为单位为伏的浮点电压值。例如，如果测得的ADC_SAR_Seq输出为0.534 V，则返回值为0.534。电压计算取决于电压参考值。当Vref基于Vdda时，用于Vdda的值针对“设计范围资源”（DWR）的System（系统）选项卡中项目设置。

参数： int16 adcCounts：ADC_SAR_Seq转换的结果。

返回值： float32:返回单位为伏的电压值

其他影响： 无

int32 ADC_SAR_Seq_CountsTo_mVolts(int16 adcCounts)

说明： 将ADC_SAR_Seq输出转换为单位为mV的16位整数电压值。例如，如果测得的ADC_SAR_Seq输出为0.534 V，则返回值为534。电压计算取决于电压参考值。当Vref基于Vdda时，用于Vdda的值针对“设计范围资源”（DWR）的System（系统）选项卡中项目设置。

参数： int16 adcCounts：ADC_SAR_Seq转换的结果。

返回值： int32：以mV为单位的电压值

其他影响： 无

int32 ADC_SAR_Seq_CountsTo_uVolts(int16 adcCounts)

说明： 将ADC_SAR_Seq输出转换为单位为mV的32位整数电压值。例如，如果测得的ADC_SAR_Seq输出为0.534 V，则返回值为534000。电压计算取决于电压参考值。当Vref基于Vdda时，用于Vdda的值针对“设计范围资源”（DWR）的System（系统）选项卡中项目设置。

参数： int16 adcCounts：ADC转换的结果。

返回值： int32：以µV为单位的电压值

其他影响： 无

void ADC_SAR_Seq_Sleep(void)

说明： 这是组件准备进入睡眠模式时的首选子程序。ADC_SAR_Seq_Sleep()例程保存当前组件状态。然后该程序调用ADC_SAR_Seq_Stop()函数，并调用ADC_SAR_Seq_SaveConfig()保存硬件配置。

在调用CyPmSleep()或CyPmHibernate()函数之前调用ADC_SAR_Seq_Sleep()函数。有关功耗管理函数的详细信息，请参考*《系统参考指南》*。

参数： 无

返回值： 无

其他影响： 无

void ADC_SAR_Seq_Wakeup(void)

说明： 该函数是将组件恢复到调用ADC_SAR_Seq_Sleep()时状态的首选子程序。ADC_SAR_Seq_Wakeup()函数调用ADC_SAR_Seq_RestoreConfig()函数以恢复配置。如果组件在调用ADC_SAR_Seq_Sleep()函数前已启用，则ADC_SAR_Seq_Wakeup()函数将重新启用组件。

参数： 无

返回值： 无

其他影响： 调用ADC_SAR_Seq_Wakeup()函数前未调用ADC_SAR_Seq_Sleep()或ADC_SAR_Seq_SaveConfig()函数可能会产生意外行为。

void ADC_SAR_Seq_Init(void)

说明： 根据自定义程序“Configure”对话框中的设置初始化或恢复组件。无需调用ADC_SAR_Seq_Init()，因为ADC_SAR_Seq_Start()子程序会调用该函数并是开始组件操作的首选方法。

参数： 无

返回值： 无

其他影响： 根据自定义程序“Configure”对话框中的内容，设置所有寄存器。

void ADC_SAR_Seq_Enable(void)

说明： 激活硬件，并开始执行组件操作。会根据时钟速度自动设置功耗。ADC_SAR_Seq_SetPower() API说明包含功耗设置与时钟速率的关系。无需调用ADC_SAR_Seq_Enable()，因为ADC_SAR_Seq_Start()子程序会调用该函数，这是开始组件操作的首选方法。

参数： 无

返回值： 无

其他影响： 无

void ADC_SAR_Seq_Disable(void)

说明： 禁用ADC_SAR_Seq组件，但不禁用ADC SAR。 此函数仅供低功耗API使用，并且用户不能调用它。

参数： 无

返回值： 无

其他影响： 无

void ADC_SAR_Seq_SaveConfig(void)

说明： 该函数会保存组件配置和非保留寄存器。它还保存Configure对话框中定义的或通过相应API修改的当前组件参数值。该函数由ADC_SAR_Seq_Sleep()函数调用。

参数： 无

返回值： 无

其他影响： 保留所有ADC_SAR_Seq配置寄存器。不会执行该函数，以备将来使用。该函数在此提供，以便API在整个组件中保持一致。

void ADC_SAR_Seq_RestoreConfig(void)

说明： 该函数会恢复组件配置和非保留寄存器。它还将组件参数值恢复为在调用ADC_SAR_Seq_Sleep()函数之前的值。

参数： 无

返回值： 无

其他影响： 调用此函数前未调用ADC_SAR_Seq_Sleep()或ADC_SAR_Seq_SaveConfig()函数可能会产生意外行为。不会执行该函数，以备将来使用。在此提供该函数，以便使API在整个组件中保持一致。

MISRA 合规性

本节介绍了MISRA-C:2004合规性和本器件的偏差情况。有两种类型的偏差，如下定义：

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

本节提供了有关器件特定偏差的信息。*《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。

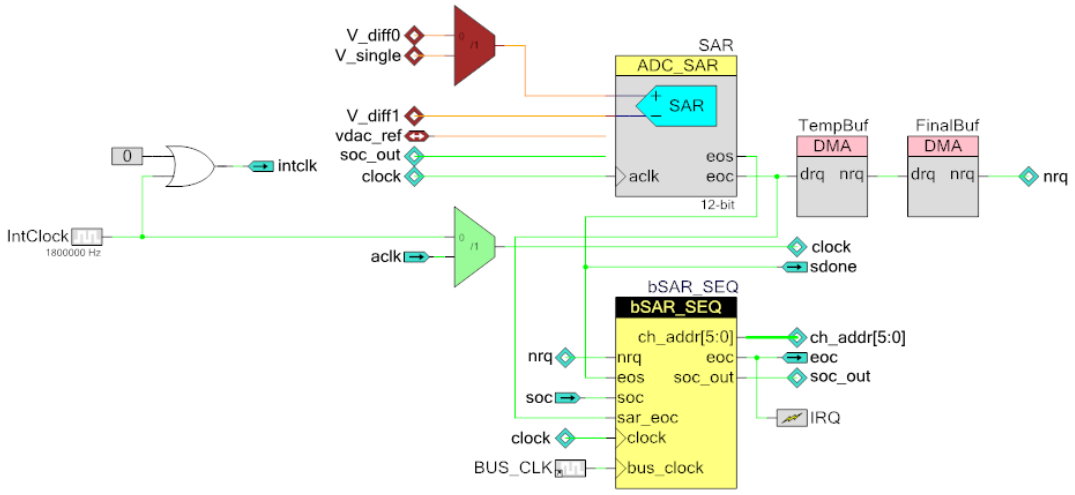
ADC_SAR_Seq 器件具有以下特定偏差：

MISRA-C：2004规则	规则类别（必须/建议）	规则描述	偏差描述
10.1	必须	某些情况下，整型表达式的数值不应隐式转换为不同的底层类型。 a) 没有转换到相同符号的更宽的整数类型， b)表达式是复合的， c)表达式不是一个常量，而是函数参数， d)表达式不是一个常量，而是返回表达式。	DMA组件提供了各个通用整数类型的定义。
13.2	建议	对一个非零值进行显性测试，除非操作数的有效的Boolean值。	DMA组件提供了对各个通用整数类型的定义。通过对这些定义进行“逻辑或”（OR）运算，可以得到各函数参数。
17.4	必须	数组索引是唯一允许的指针运算形式。	使用数组索引形式的指针运算访问DMA结构。

此组件有以下嵌入式组件，即：ADC_SAR、DMA、中断和硬件 AMUX。MISRA 合规性与特定偏差的相关信息，请参见相应器件数据手册。

功能描述

下图显示框图。ADC SAR 组件按顺序对来自硬件 AMUX 的模拟信号进行采样和转换。第一个 DMA 通道逐次将结果从 SAR ADC 转移到临时 RAM 缓冲区内。当所有通道的完整数据集均位于临时 RAM 缓冲区内时，会通过一个 DMA 脉冲段将临时 RAM 缓冲区中的内容传输到结果（SRAM）缓冲区内。由于使用了一个临时缓冲区进行收集所有的数据，因此每次扫描都会更新结果缓冲区中的内容。这样，在被下一个 DMA 覆盖前，几乎可以使用这些结果的全部扫描时间。



寄存器

状态寄存器

ADC_SAR_SEQ_STATUS_REG

位	7	6	5	4	3	2	1	0
值	未使用							EOC

- EOC — 结束转换。当完成所有通道的一个转换周期时，会设置该位。

资源

ADC_SAR_Seq 使用安装在硅片内固定模块 SAR 上相应的 SAR ADC 组件。

配置	资源类型					
	数据路径单元	宏单元	状态单元	控制/ Counter7单元	DMA通道	中断
默认值	_	45	1	2	2	1

API 存储器大小

根据编译程序、器件、所使用的 API 数量以及组件的配置情况，组件的存储器大小也不一样。下表提供默认组件配置中可用的 API 的存储器大小。每个通道使用 SRAM 中的 4 字节进行存储结果数据，因此增加通道数量会相应增大 SRAM（每个通道对应 4 字节）。

通过使用“释放”模式下的相应编译器，可以进行测量操作。在该模式中，存储器的大小得到优化。对于特定的设计，分析编译程序生成的映射文件后可以确定存储器的大小。

配置	PSoC 5LP（GCC）	
	闪存 字节	SRAM 字节
默认值	1410	37

直流和交流电气特性

下面的值表示了预计性能，它们基于初始特性数据。除非另有指定，否则运行条件为：

- Fclk = 1-18 MHz

  注意：只有总线时钟频率至少是组件时钟频率的两倍时，才能保证得到所需的采样率。

- 输入范围 = $\pm V_{REF}$
- 10 µF 的旁路电容

更多有关信息，请参考 ADC SAR 组件数据手册中的相关内容。

组件更改

本节列出了各版本的主要组件更改内容。

版本	更改内容	更改原因/影响
1.0.a	Minor datasheet edit.	
1.0	首次组件发行	

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）对以原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或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。