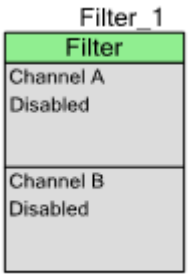
滤波器
2.0

特性



- 为在若干 PSoC 3 和 PSoC 5 器件的数字滤波器模块 (DFB) 上运行的滤波器提供便捷的用户配置
- 支持两个单独的滤波器通道，其中每个通道均构造成为单独设计的四个分级级联。
- 采用多 FIR 和 IIR（双二阶）滤波器方法（包括用户系数输入），可提供更大的灵活性
- 可提取最终系数值以供日后分析之用

概述

可定制化的滤波器组件，可允许用户将传送至数字滤波器的通道配置为一个或两个数据流通道，以便使用 DMA、中断或轮询来管理数据流。DFB 的 128 个数据和系数地址可以根据需要在两个滤波器通道之间共享使用。定制器将报告（但不设置）在用户声明的采样间隔内执行滤波时所需的最小总线时钟频率。

此组件可支持数量庞大的使用案例。如果您在使用过程中遇到任何异常情况，请及时上报 psoc_creator_fb@cypress.com，并准确描述导致该情况的具体操作，以协助赛普拉斯调查。

输入/输出连接

本节介绍滤波器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。

中断 – 输出*

在响应数据就绪事件时，如果将其中任一通道配置为生成中断，则中断输出将启用。将硬件信号连接至 ISR 组件以处理中断子程序。此信号终端仅在通道选择“中断”作为数据就绪信号时才可见。该终端为两个通道所共用。

DMA 请求 – 输出*

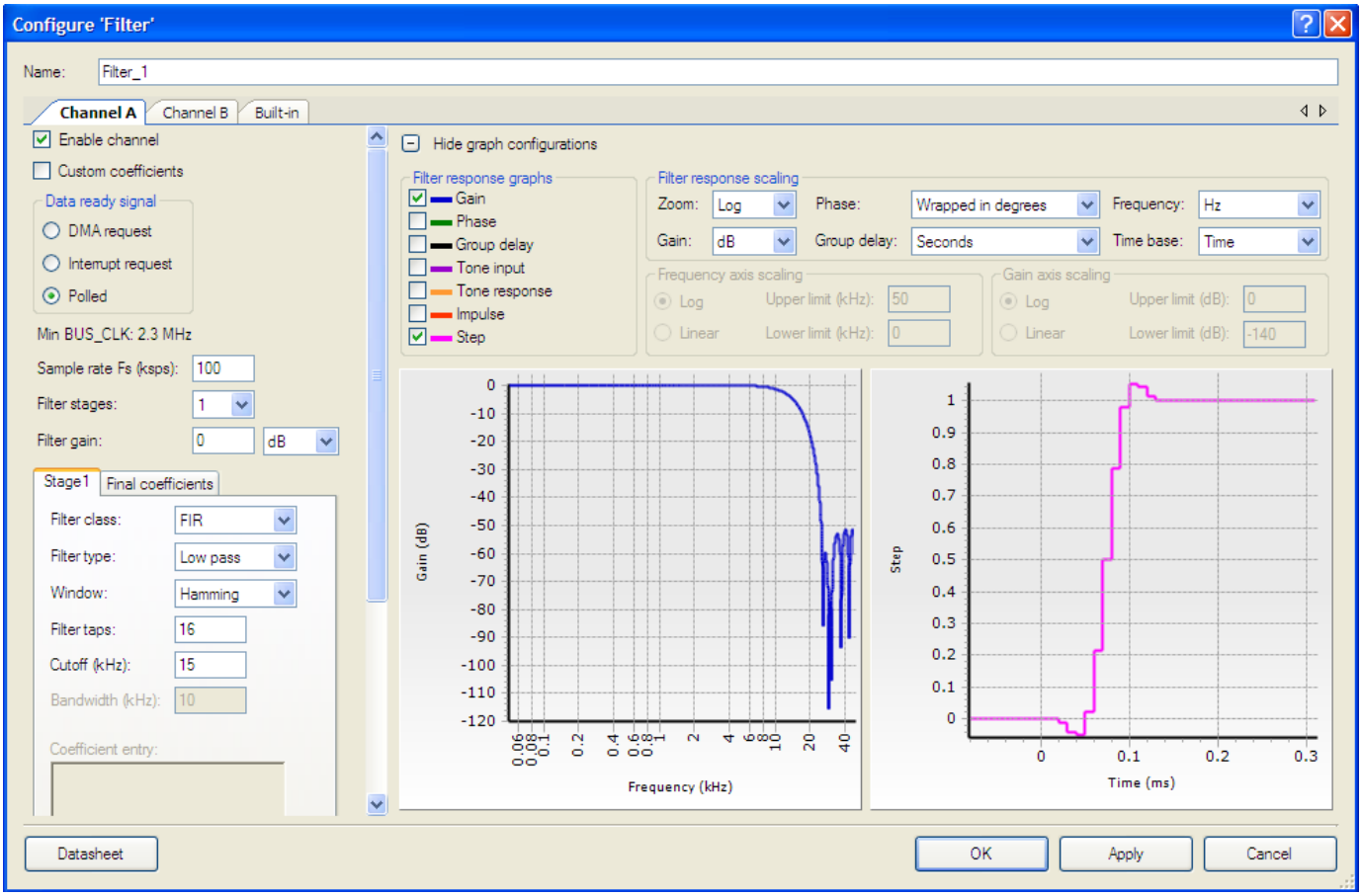
在响应数据就绪事件时，如果将其中任一通道配置为生成 DMA 请求，则系统将针对该通道启用 DMA 请求。每个通道均有其单独的 DMA 请求。将硬件信号连接至 DMA 组件以处理 DMA 子程序。

DMA 请求输出信号在读取之前将保持高电平，读取时将转为低电平。由于不提供本地缓存功能，因此用户必须确保在发送下次输出信号之前从输出信号中读取每个样本。如果在此期间未成功读出样本，则您将无法获得新的 DMA 请求输出，继而无法读取下一样本。

元件参数

将一个滤波器组件拖放到您的设计上，并双击以打开 **Configure**（配置）对话框。更改 **Name**（名称）字段以调整滤波器的实例名称。

注意 一个设计中仅应放置一个滤波器组件（或者应用 DFB 的任何其他组件）。系统不提供任何措施禁止用户安置两个组件，并且定制器将在两个组件上独立运行，但是在项目设计过程中，如果原理图中存在多个组件，设计将无法正确构建和正常运行。



滤波器参数

系统提供两组参数：

- 滤波器参数（位于左侧），可确定滤波器响应。
- 显示参数（位于右上方），可确定显示哪些滤波器响应图表以及这些图表的缩放方式。

启用通道

此参数用于启用或禁用该通道的代码生成功能。仅当各个通道滤波器的资源充足时才可启用该通道。如果没有足够的资源用来支持配置更改，则系统将显示警告图标。由于对此错误状态的跟踪十分复杂，因此您可能需要多次来回单击不同设置才可清除所有错误标记。此外，错误标记的消失并不保证不存在任何错误。

定制系数

此参数用于启用或禁用通道的定制系数输入。如果已启用此参数，则可以在 **Coefficient entry**（系数输入）文本框中输入定制系数。这些定制系数可用于计算滤波器响应以及生成 DFB 微代码。在滤波器 2.0 中，定制设置适用于滤波器的所有分级；即对按组件本身设计的分级使用定制系数可确保不会混淆各个分级。禁用此参数后，定制器将根据用户要求计算滤波器系数，且 **Coefficient entry**（系数输入）框将保留为空。该通道全部四个分级汇总所得的系数为此设计过程的最终系数，此数值将显示在 **Final coefficients**（最终系数）抽头中。使用剪贴板可以将这些系数粘贴复制到文本文件或电子表格中，以备日后处理。将其贴回至定制系数模式下的系数输入框，可在定制设计中启用内部设计滤波器。

当启用此参数时，**Filter stages**（滤波器分级）参数（除 **Filter class**（滤波器类）和 **Filter taps**（滤波器抽头）外）均为无效，因此这些参数为禁用状态。禁用此参数可重新启用 **Filter stage**（滤波器分级）参数。在 FIR 定制模式下，定制器将新线路上的各个数字输入解析为抽头值（零表示已接受）并计算抽头的总个数。在定制双二阶模式下，定制器预期的条目数为五的倍数，当实际情况有出入时该程序将执行错误报告。此外，它还检查双二阶分母系数的稳定性；如果一个或多个双二阶出现不稳定，其系数将不予接受。

数据就绪信号

此模块可配置为，在数据准备就绪时是通过特定于各个通道的 DMA 请求信号，还是两个通道之间共享的中断请求提醒用户。另外，您还可以轮询状态寄存器来查看 DFB 输出中是否存在新数据。但是，在开始 DFB 操作之前必须在 INT_CTRL 寄存器中启用中断，方可在此处对其轮询。

有关 INT_CTRL 和 SR（状态寄存器）寄存器的详细信息，请参阅寄存器一节。由于输出保持寄存器为双缓冲，因此您必须确保及时读取数据，以免其被覆盖。

如果您正以高实时采样率运行滤波器，则 DMA 也许是速率与之相匹配的唯一方法。由于不涉及主 CPU，因此 DFB 可以通过具有 DSP 扩展指令集的微控制器以尽可能大的采样率实施复杂滤波器。

采样率

向 **Sample rate**（采样率）框（始终以 ksps 为单位）输入的比率不会影响硬件运行，仅影响滤波器系数的设计处理以及显示结果所用的图形缩放。您必须以所需物理采样率通过 DFB 抽取数据。定制器将声明采样率限制在 1 sps 至 10 Msps 的范围内；此限制仅为方便您的报告，因为 PSoC 3 或 PSoC 5 器件的最大 bus_clk 频率不会超过 80 MHz。

由于滤波器 2.0 中不存在抽取或差值，因此多级滤波器各个分级的采样率相同。对于两个通道而言，采样率无需相同，但是当两个采样率相同或特别相近时，滤波器 2.0 的 DFB 固件操作效率最高。

在完成滤波器设计输入配置之后，滤波器 2.0 将计算 DFB 运行所需的最低 PSoC 3/PSoC 5 bus_clk 频率，以确保所有采样均能正常处理。当此值超过系统中的最高可用时钟频率时，将无法实现滤波器。不提供缓冲；这就意味在较快的通道中，数字滤波器模块代码必须能够在两个来样间隔期间内执行*两个*滤波器。

如果您知道系统的最大可用 bus_clk 频率、两个通道各自的系数计数和滤波器极点，则可以计算滤波器运行时的最大采样率。如果两个通道均在使用，则此频率应为具有较高采样率的通道（如果两个通道运行速率相同，则为两个通道）运行时的频率。可能的最大采样率通过将最高可用 bus_clk 频率除以两个滤波器所需的 DFB 周期数计算得来。因此，每个通道的周期计数计算方法如下：

仅 FIR：	10 + 抽头数
仅双二阶偶数阶次总计：	13 + 5 × 阶次
仅双二阶奇数阶次总计：	18 + 5 × 阶次
双二阶偶数阶次总计和 FIR：	20 + 抽头数 + 5 × 阶次
双二阶奇数阶次总计和 FIR：	25 + 抽头数 + 5 × 阶次

将针对两个通道计算所得的结果相加。即可得到执行两个通道所需的 bus_clk 周期总数。除以该值则得到您计划使用的 bus_clk 频率，同时得到两个通道之中较快的通道运行时要用的采样率数值。

滤波器分级

每个通道均由最多四个不同的级联滤波器分级；该数值可通过 **Filter stage**（滤波器分级）下拉列表进行选择。如果系统中没有适当的资源用以支持添加另一分级，则系统将发出警告通知。每次输入新参数时，滤波器响应图表将更新显示完整级联。您可以通过其各自的抽头访问各个分级。

您也可以单独配置各个滤波器分级的参数。例如，FIR 低通滤波器可以与双二阶陷波滤波器串联。无论采样哪种输入顺序，定制器均将以相同的方式集成全部四个分级的系数和极点。将所有 FIR 分级卷积可得到一个 FIR 滤波器，而全部分二阶分级可作为一个串联予以实施。通过一套相当复杂的动态范围管理算法检查所有分级的顺序并调整阶次和内部增益，以使滤波器实现最高质量的信号处理。

滤波器增益

您可以通过此参数为滤波所得的信号提供直流增益。此参数以线性和分贝两种形式表示增益值。线性增益范围应为 0.01 至 100；相应的分贝增益应为 –40 dB 至 40 dB。如果您要设置的增益超出此范围，则定制器将标记错误。如果必要，您可以为线性增益输入负数以反转响应。此信号将不在分贝屏幕上显示。

请注意，由于大部分滤波器响应的某些信号会出现时域过冲，因此为确保输出结果不会溢出，最佳做法是以低于单位的增益运行滤波器。默认增益为单位增益，但是在输入信号可能会超出数字全量程的典型设计中，赛普拉斯建议将增益选设为 0.25x (– 12.04dB)。设置定制器中的滤波器设计子程序以在选中此增益之后，确保在输入任何可行的数字信号时滤波器的任何内部分级均不会过载。

Final coefficients（最终系数）抽头

除了用于设置滤波器分级的抽头，**Final coefficients**（最终系数）抽头中还包含将在滤波器中使用的最终系数的显示窗口。其中包括由定制器为优化性能而执行的所有系数舍入、增益调整和双二阶重新排序。您可以右键单击此窗口选择并复制相关内容。然后将这些内容复制至电子表格中，以供日后分析之用。滤波器分级：系数输入窗口 一节讨论在输入和输出 FIR 和 IIR 滤波器的系数时所使用的格式。

滤波器分级：滤波器类

可选选项包括 **FIR** 和 **Biquad**（双二阶）。双二阶滤波器作为二阶滤波段的串联级联实现。奇数阶传递函数中出现的一阶滤波段作为 z^{-2} 系数为零的双二阶实现。

滤波器分级：滤波器类型 (FIR)

对于 FIR 滤波器，可用选项包括**低通**、**高通**、**带通**、**带阻**、**Sinc^4** 和 **Hilbert**（希尔波特）滤波器。

Sinc^4 滤波器是一款通带逐渐增大的专用低通滤波器。该款滤波器专门用于补偿使用四阶 sinc 抽取滤波器的 delta-sigma ADC（如以 16 位或更低的分辨率运行的 PSoC 3 和 PSoC 5 中的 ADC）的迟缓频率响应。

Hilbert 滤波器是一款特殊的、可产生 90 度额外相移的高通滤波器。该款滤波器可用于某些通讯信号处理。

对于双二阶滤波器，可用选项包括**低通**、**高通**、**带通和带阻**。

滤波器分级：窗口 (FIR)

系统提供了多种使用 FIR 滤波器时的窗口设定方法。您可以根据需求调整它们之间的差异。不同的窗口设定方法对通带波纹、过渡带宽和阻带衰减的影响也有所不同。

- **矩形**：此方法表示未设定范围；理想低通滤波器的 sinc 脉冲响应在超出抽头数范围时将截断至零。这些滤波器将呈现较大的通带波纹、快速衰减以及以及较弱的阻带衰减。由于来自 Gibbs（吉普斯）现象的较大波纹效应，因此很少使用此窗口。
- **Hamming**（汉明）：在滤波器 2.0 中，基于 Albrecht 该窗口实际上作为稍微改进的 Hamming 窗口使用。它将呈现出一种略微平坦但更均匀的阻带。这是一款优质的通用 FIR 滤波器，是放置新组件的默认选项。
- **Blackman**（布兰克曼）：一款改进的 Blackman 窗口接近于 Blackman-Nuttall（布莱克曼–纳托尔），同样基于 Albrecht）。该窗口与 Hamming 类窗口相比可以提供更大的阻带衰减，但其过渡带也更宽。

滤波器分级：滤波器抽头 (FIR)

此版输入框可用于 FIR 类滤波器分级。当仅使用 FIR 滤波器时，所有滤波器合并后的总体积可达 128 抽头。FIR 滤波器的阶次等于抽头数减去一。

滤波器分级：形状(二分阶）

在滤波器 2.0 中使用二分阶类滤波器时，可用形状包括 Butterworth（巴特沃斯）、Bessel（贝塞尔）和 Chebyshev（切比雪夫）三种。注意：滤波器 2.0 中的贝塞尔实现采用经典线性版中的双二线变形技术，因此，对于占采样率重要部分的截止频率来说不会保持平缓波群延迟行为。

滤波器分级：阶次（双二阶）

此版输入框可用于双二阶类滤波器分级。低通和高通滤波器可以是奇数，也可以是偶数，但是，由于 DFB 采用相同的分二阶拓扑结构实施单个极点，因此实施时总是四舍五入到最接近的偶数。带通和带阻滤波器仅可能是奇数阶，如果在此情况下输入偶数，系统将会发出错误警报。切比雪夫 (Chebyshev) 和巴特沃斯 (Butterworth) 滤波器的最大阶次为 50。当处于最大阶次且带通或带阻带宽较窄时，定制器中的数字限制可能会产生意外后果。

设计贝塞尔低通和高通滤波器时，其滤波器最大阶次为 25。

当 N 为偶数时，N 阶串联分二阶滤波器所需的内存位置数为 2.5 N；如果 N 为奇数，则其内存位置数为 2.5(N + 1)。分二阶滤波器要求三个内部内存位置来承载其他变量，因此，使用分二阶或分二阶段 + FIR 时，所有滤波器合并后的总体积不能超过 125 内存位置。

滤波器分级：截断

为 Sinc^4、低通、高通和 Hilbert 滤波器输入通带边缘频率。滤波器在此频率下的预期增益视滤波器类别和类型而定。对于 FIR 低通、高通和 Hilbert 滤波器，在所输截止频率下其响应名义上应为 –6 dB。对于双二阶贝塞尔和巴特沃斯滤波器，响应为 –3 dB。对于契比雪夫滤波器，相对于通带中的最高增益，其响应等于所输入的波纹值。

FIR sinc^4 滤波器的增益响应将通过实验进行最佳评估。

滤波器分级：中心频率和带宽（带通和带阻）

带通滤波器的带宽是符合针对滤波器类别和类型给定的截断标准的高低频率之间的频率差。中心频率位于高频和低频之间，但是其与这两个频率之间的确切关系取决于滤波器类型和类别。

对于 FIR 类滤波器，带通或带阻滤波器的中心频率是高截止频率和低截止频率的算术平均值。也就是说，客户创建的响应围绕中心频率呈算术对称性。中心频率和带宽均为 10 kHz 的 FIR 带通或带阻滤波器在 5 kHz 和 15 kHz 时均为 –6-dB。

滤波器 2.0 中所用的带通和带阻转换类型可提供呈几何对称的频率响应。双二阶类带通滤波器由其高低截止频率进行确定，其高低截止频率围绕用户输入的中心频率呈*算术*对称分布，此情况与使用 FIR 滤波器时相同。因此，如果针对巴特沃斯双二阶带通滤波器输入的中心频率和带宽均为 10 kHz，则在 5 kHz 和 15 kHz 时点均为 –3-dB。此类滤波器的“真实”中心频率并非 10 kHz；在本例中应为 SQRT(5 kHz × 15 kHz) = 8.66 kHz。

对于双二阶类带阻滤波器，其设计旨在输入中心频率时拥有最大衰减。这意味着无法以 FIR 相同的情况定位通带截止频率。实际上，要获得截止频率，需要进行一些简单的计算：

$$F_{lower} = SQRT(0.25 \times BW^2 + F_{center}^2) – 0.5 \times BW$$

因而：$F_{upper} = F_{lower} + BW$

在上述示例中，当中心频率和带宽均为 10 kHz 时，此计算为 F_{lower} = 6.18 kHz 且 F_{upper} = 16.18 kHz。

滤波器分级：波纹 (IIR Chebyshev)

此参数仅对双二阶类切比雪夫滤波器有效。它可用于确定滤波器的理论通带波纹。其允许范围为 0.00001 dB 至 3 dB。

滤波器分级：系数输入窗口

选定 **Custom Coefficients**（定制系数）复选框，然后使用此文本框在选定的滤波器分级中输入定制系数；请参见定制系数。

在连续行中输入各系数作为浮点值。空格可忽略，但数值零为有效项。滤波器 2.0 中不接受非数值项。

对于双二阶滤波器，首先输入三个分子系数（z^0、z^{-1} 和 z^{-2}），然后输入两个分母系数（假设 z^0 分母系数为单位）。验证所输入系数的有效性，如果证实任一系数为无效系数（或总数不被五整除），则将显示错误指示符。此外，还应测试所输入分母系数的稳定性。如果测得任何分二阶不稳定，则将显示警告。分二阶系数还可应用于针对内部生成的系数所使用的相同增益调整和排序算法，因此其实现顺序可能与输入顺序相同。此外，滤波器通带中的峰值增益将调整为用户增益框中输入的数值。您可以输入任何数值的分子刻度，该算法将对其进行适当的调整。如果您要使

用滤波器 2.0 实施 PID 或其他控制回路，则需要计算您自己的增益值并输入增益框，以确保总增益符合您的需要。

对于 FIR 滤波器，输入的第一个值作为 z^0 的系数，即无延迟抽头。对于仅含 FIR 的滤波器，不使用任何单独的增益调整算法。在 FIR 滤波器中，系数值的最大允许范围为 –1 至 1-2-23。要输入该范围之外的数值，您必须具备 Filter Gain（滤波器增益）值，以便调降这些数值以使其符合该范围。当所输系数无效时，将提供此增益值。

当 FIR 和双二阶组合使用时，系统将在实施双二阶之前针对 FIR 部分自动应用增益调整。您可以在 **Final coefficients**（最终系数）抽头中查看整个级联的调整结果。您还可以根据需要将该视图中的结果粘贴到另一个应用程序中，以备日后分析之用。

显示参数

显示参数仅影响滤波器响应在配置窗口中的显示方式。而对代码生成或滤波器设置无任何影响。

通过此区域顶部的图形配置按钮可以显示或隐藏设置参数选择；该操作使图形更易于读取。请注意，当多个窗格显示在图形区域中时，由于自动绘制子程序的限制，绘制轴可能不会按预期进行编号。

各绘图均由频率参数和时间参数两个子绘图组成。右键单击位于剪贴板侧面、以位图格式显示的各个子绘图副本，以便将其粘贴至您自己的报告中。

滤波器响应图形

增益 — 启用后，显示高于频率的滤波器总响应的幅度。

相位— 启用后，显示高于频率的滤波器总响应的相移。

波群延迟— 启用后，显示高于频率的滤波器总响应的群波延迟。

谐频输入 — 启用后，显示可用作滤波器输入的滤波器中心频率或阶段频率处的正弦波信号。如果使用多重滤波分级，则谐频等于中心频率或截止频率的平均值。在滤波器 2.0 中，无法单独设置此正弦波的频率。

谐频响应— 启用后，显示滤波器对预定谐频输入波形的响应（请参阅谐频输入 ）。

脉冲 — 启用后，显示滤波器对正向单一样本脉冲的响应。

步进 — 启用后，显示滤波器对正向单位步进函数的响应。

滤波器响应调整：缩放。

可用选项为 **Log**（对数）、**Linear**（线性）和 **Custom**（定制）。

对数缩放选项可提供基于对数频率刻度（从直流到奈奎斯特频率）的滤波器响应视图。

线性选项提供从直流到奈奎斯特频率的通带频率线性频率刻度。

选择定制选项可为增益和频率启用最大和最小限值输入设置。您可以在 Custom（定制）视图中定义频率和增益轴限值。在此视图中，频率和增益轴在线性模式和对数模式下均可设置。

滤波器响应调整：增益

选择 **dB** 增益可以分贝形式显示增益响应的增益值。选择 **Linear**（线性）可以线性刻度显示增益值。

滤波器响应调整：相位

通过此参数，可以选择查看以弧度或度显示的卷绕相位或解卷绕相位。可用选项包括 **Unwrapped in degrees**（解卷绕度）、**Wrapped in degrees**（卷绕度）以及 **Unwrapped in radians**（解卷绕弧度）和 **Wrapped in radians**（卷绕弧度）。注意：对于某些带有传输零点的滤波器（如带阻滤波器），其相位响应图不连续，即使在选择解卷绕相位的情况下也是如此。

滤波器响应调整：波群延迟

通过此参数，可以选择 Group Delay（群组延迟）响应作为采样时间（以微秒为单位）或采样数量。

滤波器响应调整：频率

通过此参数，可以选择 Frequency（频率）响应 X 轴作为采样值（以 kHz 为单位）或采样数量。

滤波器响应调整：时基

通过此参数，可以选择 Time（时间）响应 X 轴作为以毫秒或采样计数为单位的值。

频率轴调整：对数/线性选择

此选项仅在选择 **Custom**（定制）放大时才有效。选择 **Log**（对数）可按对数刻度设置“频率”响应。选择 **Linear**（线性）可按线性刻度设置 X 轴。

频率轴调整：上限

此选项仅在选择 **Custom**（定制）放大时才有效。通过该选项，可为频率响应图的 X 轴设置上限。最大上限应小于或等于采样率的二分之一。

频率轴调整：下限

此选项仅在选择 **Custom**（定制）放大时才有效。通过该选项，可为频率响应图的 X 轴设置下限。下限应小于上限值且不得小于零。

增益轴调整：对数/线性选择

此选项仅在选择 **Custom**（定制）放大的情况下有效。选择 **Log**（对数）可按对数刻度设置增益轴。选择 **Linear**（线性）可按线性刻度设置增益轴。

增益轴调整：上限

此选项仅在选择 **Custom**（定制）放大时才有效。通过该选项，可以按 dB 或线性任一刻度设置增益响应上限，具体情况视您选择 **Log**（对数）还是 **Linear**（线性）按钮而定。

增益轴调整：下限

此选项仅在选择 **Custom**（定制）放大时才有效。通过该选项，可以按 dB 或线性任一刻度设置增益响应下限，具体情况取决于您选择 **Log**（对数）还是 **Linear**（线性）按钮。下限值应低于上限值。

放置

滤波器 2.0 组件将使用整个 DFB，因此在一个运转的项目中，仅能放置一个滤波器（或者要求 DFB 的任何其他组件）。如果放置了多个滤波器组件，每个滤波器均可通过其自有的定制器进行设置并保存相关属性。因此，在构建项目之前，可以在初始原理图中包含多个滤波器，将其作为一种保存设置的方式。

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	
不可用	不可用	不可用	不可用	不可用	不可用	3116	69	1

除 DFB 之外，滤波器 2.0 组件不耗用其他任何资源。为实现最大吞吐量，您将可能使用 DMA 来管理数据。

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与组件交互。下表列出并描述了每个函数的接口以及“include”文件提供的相关常量。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“Filter_1”分配给指定设计中组件的第一个实例。您可以将其重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“Filter”。

函数	说明
Filter_Start()	配置和启用滤波器组件硬件以进行中断、DMA 和滤波器设置。
Filter_Stop()	停止滤波器运行，并使硬件断电。
Filter_Read8()	读取滤波器输出保持寄存器上的当前值。最高有效字节的单字节读数。
Filter_Read16()	读取滤波器输出保持寄存器上的当前值。最高有效字节的双字节读数。
Filter_Read24()	读取滤波器输出保持寄存器上的当前值。最高有效字节的三字节读数。
Filter_Write8()	向滤波器输入分级寄存器写入新的 8 位样本。
Filter_Write16()	向滤波器输入分级寄存器写入新的 16 位样本。
Filter_Write24()	向滤波器输入分级寄存器写入新的 24 位样本。
Filter_ClearInterruptSource()	向状态寄存器写入 Filter_ALL_INTR 掩码，以清除任何活动中断。
Filter_IsInterruptChannelA()	识别通道 A 是否已触发数据就绪中断。
Filter_IsInterruptChannelB()	识别通道 B 是否已触发数据就绪中断。
Filter_Sleep()	停止并保存用户配置。
Filter_Wakeup()	恢复并启用用户配置。
Filter_Init()	初始化或恢复默认滤波器配置。
Filter_Enable()	启用滤波器。
Filter_SaveConfig()	保存滤波器非保留寄存器的配置。
Filter_RestoreConfig()	恢复波器非保留寄存器的配置。
Filter_SetCoherency()	在连贯寄存器中设置关键连贯字节。

全局变量

变量	说明
Filter_initVar	指示滤波器是否已初始化。变量将初始化为 0，并在第一次调用 Filter_Start() 时设置为 1。这样，第一次调用 Filter_Start() 子程序后，组件不用重新初始化即可重启。

	如果组件需要重新初始化，则在调用 Filter_Start() 或 Filter_Enable() 函数之前可以先调用 Filter_Init() 行数。

定义

- **Filter_CHANNEL_x —** Filter_CHANNEL_A 或 Filter_CHANNEL_B。指定函数调用操作发生通道时使用。
- **Filter_CHANNEL_x_INTR —** 状态寄存器的 CHANNEL_A 或 CHANNEL_B 中断掩码。
- **Filter_ALL_INTR —** 状态寄存器的可能中断掩码。

void Filter_Start(void)

说明： 这是开始执行组件操作的首选方法。配置和启用滤波器组件硬件可进行中断、DMA 和滤波器设置。

参数： 无

返回值： 无

副作用： 无

void Filter_Stop(void)

说明： 使滤波器硬件停止运行并断电。

参数： 无

返回值： 无

副作用： 无

uint8 Filter_Read8(uint8 channel)

说明： 读取通道 A 或通道 B 输出保持寄存器的最高阶次字节。

参数： uint8 通道：应读取的滤波器通道。包括 Filter_CHANNEL_A 和 Filter_CHANNEL_B 两个选项。

返回值： 8 位滤波器输出值

副作用： 无

uint16 Filter_Read16(uint8 channel)

说明： 读取通道 A 或通道 B 输出保持寄存器的两个最高阶次字节。

参数： uint8 通道：应读取的滤波器通道。包括 Filter_CHANNEL_A 和 Filter_CHANNEL_B 两个选项。

返回值： 16 位滤波器输出值

副作用： 无

uint32 Filter_Read24(uint8 channel)

说明： 读取通道 A 或通道 B 输出保持寄存器的全部三个。

参数： uint8 通道：应读取的滤波器通道。包括 Filter_CHANNEL_A 和 Filter_CHANNEL_B 两个选项。

返回值： 24 位符号扩展输出值的 32 位无符号表示

副作用： 无

void Filter_Write8(uint8 channel, uint8 sample)

说明： 写入通道 A 或通道 B 输入分级寄存器的最高阶次字节。

参数： uint8 通道：应读取的滤波器通道。Filter_CHANNEL_A 和 Filter_CHANNEL_B。

uint8 样本：要写入输入寄存器的值

返回值： 无

副作用： 此函数仅写入最高有效字节。如果尚未将最低阶次字节设置为零，这可能会导致噪声增加输入样本中。

void Filter_Write16(uint8 channel, uint16 sample)

说明： 写入通道 A 或 通道 B 的输入分级寄存器的两个最高有效字节。

参数： uint8 通道：应读取的滤波器通道。包括 Filter_CHANNEL_A 和 Filter_CHANNEL_B 两个选项。

uint16 样本：要写入输入寄存器的值

返回值： 无

副作用： 无

void Filter_Write24(uint8 channel, uint32 sample)

说明： 写入通道 A 或通道 B 输入分级寄存器的全部三个字节。

参数： uint8 通道：应读取的滤波器通道。包括 Filter_CHANNEL_A 和 Filter_CHANNEL_B 两个选项。

uint32 样本：要写入输入寄存器的值

返回值： 无

副作用： 无

void Filter_ClearInterruptSource(void)

说明： 将 Filter_ALL_INTR 掩码写入状态寄存器以清除任何活动中断。有关此掩码的定义，请参阅之前的定义一节。

参数： 无

返回值： 无

副作用： 无

uint8 Filter_IsInterruptChannelA(void)

说明： 识别通道 A 是否已触发数据就绪中断。

参数： 无

返回值： 如果通道 A 无任何中断，为 0；否则，为正值。

副作用： 无

uint8 Filter_IsInterruptChannelB(void)

说明： 识别通道 B 是否已触发数据就绪中断。

参数： 无

返回值： 如果通道 B 无任何中断，为 0；否则，为正值。

副作用： 无

void Filter_SetCoherency(uint8 channel, unit8 byte_select)

说明： 设置 DFB 连贯寄存器中的值。此值用于确定关键连贯字节。当需要更新字段时，软件通过关键连贯字节“告诉”硬件最后写入或读取字段中的哪个字节。

参数： uint8 通道：包括 Filter_Channel_A 和 Filter_Channel_B 两个选项。

uint8 byte_select：确定将三个字节中的哪个字节设置为关键连贯字节。包括高位字节、中位字节和低位字节三个选项。

返回值： 无。

副作用： 默认情况下，高位字节为关键连贯字节。使用此 API 更改默认设置之后，再使用上文所述 Filter_Read() 和 Filter_Write() API 可能会导致意外行为。

void Filter_Sleep(void)

说明： 停止 DFB 操作。保存配置寄存器和组件启用状态。应当刚好在进入睡眠之前调用。

参数： 无

返回值： 无

副作用： 滤波器输出寄存器为非保留寄存器，在其进入睡眠后将不保存上述参数。因此，请在进入睡眠之前确保无任何待处理的转换。

void Filter_Wakeup(void)

说明： 这是将组件恢复到调用 Filter_Sleep() 状态时的首选 API。Filter_Wakeup() 函数调用 Filter_RestoreConfig() 函数以恢复配置。如果组件在调用 Filter_Sleep() 函数前已启用，则 Filter_Wakeup() 函数也将重新启用组件。

参数： 无

返回值： 无

副作用： 调用 Filter_Wakeup() 函数前未调用 Filter_Sleep() 或 Filter_SaveConfig() 函数可能会产生意外行为。

void Filter_Init(void)

说明： 根据定制器“配置”对话框设置来初始化或恢复组件。无需调用 Filter_Init()，因为 Filter_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： 所有寄存器将复位至初始值。这将重新初始化组件。

void Filter_Enable(void)

说明： 激活硬件并开始执行组件操作。您无需调用 Filter_Enable()，因为 Filter_Start() API 会调用此函数，该函数是开始执行组件操作的首选方法。

参数： 无

返回值： 无

副作用： 无

void Filter_SaveConfig(void)

说明： 此函数会保存组件配置和非保留寄存器。此函数还将保存当前 Configure（配置）对话框中定义的或通过相应 API 修改的组件参数值。该函数由 Filter_Sleep() 函数调用。

参数： 无

返回值： 无

副作用： 无

void Filter_RestoreConfig(void)

说明： 此函数恢复组件配置和非保留寄存器。它还将组件参数值恢复为调用 Filter_Sleep() 函数之前的值。

参数： 无

返回值： 无

副作用： 调用此函数之前未调用 Filter_SaveConfig() 或 Filter_Sleep() 将导致意外行为

固件源代码示例

PSoC Creator 在 Find Example Project（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 **File**（文件）菜单中的对话框。根据需要，使用对话框中的 **Filter Options**（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

该滤波器组件用于生成 DFB 协同处理器的所需代码并配置滤波器组件。多级滤波器通过卷积算数合并成为一个滤波器，从而产生一个对所有通道而言都相对较大的滤波器。未来模式中还将增加对通过每个通道的 IIR-FIR 流的支持。

DMA

DMA 组件可用于从滤波器输出寄存器向 RAM 传输转换结果。DMA 组件也可用于向滤波器输入寄存器传输输入样本值。滤波器数据就绪信号应与 DMA 的数据请求信号相关联。使用 DMA 向导可按如下表所示配置 DMA 操作。

滤波器分辨率	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
8 位	Filter_HOLDBH_PTR	源	DMA_Req_B	电平	接收来自滤波器输出寄存器 Filter_HOLDB 的 8 位滤波输出值
	Filter_HOLDAH_PTR	源	DMA_Req_A	电平	接收来自滤波器输出寄存器 Filter_HOLDA 的 8 位滤波输出值
	Filter_STAGEAH_PTR	目标	DMA_Rea_A 或不可用	电平或不可用	接收从 RAM 到滤波器输入寄存器 Filter_STAGEA 的 8 位输入样本值
	Filter_STAGEBH_PTR	目标	DMA_Req_B 或不可用	电平或不可用	接收从 RAM 到滤波器输入寄存器 Filter_STAGEB 的 8 位输入样本值。
16 位	Filter_HOLDB_PTR	源	DMA_Req_B	电平	接收来自滤波器输出寄存器 Filter_HOLDB 的 16 位滤波输出值
	Filter_HOLDA_PTR	源	DMA_Req_A	电平	接收来自滤波器输出寄存器 Filter_HOLDA 的 16 位滤波输出值
	Filter_STAGEA_PTR	目标	DMA_Rea_A 或不可用	电平或不可用	接收从 RAM 到滤波器输入寄存器 Filter_STAGEA 的 16 位输入样本值。
	Filter_STAGEB_PTR	目标	DMA_Req_B 或不可用	电平或不可用	接收从 RAM 到滤波器输入寄存器 Filter_STAGEB 的 16 位输入样本值
24 位	Filter_HOLDB_PTR	源	DMA_Req_B	电平	接收来自滤波器输出寄存器 Filter_HOLDB 的 24 位滤波输出值
	Filter_HOLDA_PTR	源	DMA_Req_A	电平	接收来自滤波器输出寄存器 Filter_HOLDA 的 24 位滤波输出值
	Filter_STAGEA_PTR	目标	DMA_Rea_A 或不可用	电平或不可用	接收从 RAM 到滤波器输入寄存器 Filter_STAGEA 的 24 位输入样本值
	Filter_STAGEB_PTR	目标	DMA_Req_B	电平或	接收从 RAM 到滤波器输入寄存器

滤波器分辨率	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
			或不可用	不可用	Filter_STAGEB 的 24 位输入样本值

寄存器

分级

滤波器组件的两个通道各自拥有一个专用的 24 位输入分级寄存器。在未处理数据时，滤波器将进入等待状态，在该状态下，滤波器将等至向任意寄存器写入数据之后再开始新的数据通过。

保持

在完成输入数据处理之后，最新的输出样本将保存至 24 位输出保持寄存器中。存在三种与准备输出样本系统通知有关的选项：中断、DMA 请求或轮询。

数据对齐 (Filter_DALIGN) 寄存器

DFB 要求输入数据和交付的输出结果都与 MSB 对齐。为便于系统软件操作，DFB 硬件在输入分级寄存器和输出保持寄存器中均提供有数据对齐功能。

两个分级和两个保持寄存器均支持字节访问，这解决了八位或更低位的输入和输出采样对齐问题。同理，全部四个寄存器都可映射为 32 位寄存器（仅使用四分之三的字节），因此对于 17 和 24 位之间的样本不存在任何对齐问题。但是，对于 9 和 16 之间的样本大小，由于这些样本源自或灌入在保持/分级寄存器的 (23:8) 位上，因此它们能够很容易地在总线位 (15:0) 上读/写这些样本。

数据对齐寄存器位可提供对齐功能，从而允许系统总线位 (15:0) 既可以取源于保持寄存器位 (23:8) 也可以灌入分级寄存器位 (23:8)。每个分级和保持寄存器均可通过 DALIGN 寄存器中的某个位来单独配置。如果该位被设为高电平，则将发生有效字节移位。

示例：如果 Delta-sigma ADC 的输出样本为 12 位宽且与 ADC 输出采样寄存器中的位 23 对齐，此时，如果您想将此值串流至 DFB，则可以启用 ADC 的数据对齐功能。通过此操作，可在总线位 (15:0) 上读取 ADC 输出采样寄存器的 16 位。为 A 分级输入寄存器设置 DFB 对齐功能时，可以在总线位 (15:0) 上写入此 16 位，在必要时也可将其写入 A 分级寄存器的位 (23:8)。

连贯 (Filter_COHER) 寄存器

连贯指 DFB 中包含的硬件在寄存器字段宽于总线访问宽度时保护模块，使其避免发生故障。当字段部分写入/读取（不连贯）时，此情况会产生时间间隔。

连贯检查选项可在连贯寄存中启用。此硬件可提供对分级和保持寄存器的连贯检查。

分级寄存器具有写保护，因此当通过系统软件部分升级时，基础硬件将不使用字段。保持寄存器具有写保护，因此当通过系统软件或 DMA 部分读取时，基础硬件将不更新字段。基于不同的模块配置，并非所有分级和保持寄存器字节都是必需的。连贯方法允许输入任意大小的输出字段并对其进行适当地处理。

此寄存器的位字段用于从分级和保持寄存器的三个字节中选择将哪一字节用作关键连贯字节。在连贯寄存器中，启用连贯并选择关键连贯字节。当需要更新字段时，用户（软件）通过关键连贯字节“告诉”硬件最后写入或读取字段中的哪个字节。

滤波器寄存器和 DMA 向导设置

滤波器分辨率	每次突发字节 *	Filter_COHER 和 Filter_DALIGN 的寄存器设置	字节序标志*	
			PSoC 3	PSoC 5
8 位	1	Filter_COHER_REG = 0xAA Filter_DALIGN_REG = 0x00	禁用	禁用
16 位	2	Filter_COHER_REG = 0x00 Filter_DALIGN_REG = 0x0F	TD_SWAP_EN | TD_INC_DST_ADR	禁用
24 位	4	Filter_COHER_REG = 0xAA Filter_DALIGN_REG = 0x00	TD_SWAP_EN | TD_SWAP_SIZE4 | TD_INC_DST_ADR	TD_SWAP_SIZE4

* 在 DMA 向导工具中设置。

Filter_SR_REG

滤波器状态寄存器。读取该寄存器可了解中断源。使用 Filter_ClearInterruptSource() 宏可将其清空。

位	7	6	5	4	3	2	1	0
值	INTR SEM2	INTR SEM1	INTR SEM0	INTR HOLDING REG B	INTR HOLDING REG A	RND MODE	SAT MODE	RAM SEL

此寄存器包含五个指示模块生成中断的状态位和三个路径单元状态位。

注意：如果系统软件要轮询事件但不生成中断，则必须在 INT_CTRL 寄存器中启用中断才可在此处轮询。

- 位 7：信号 2 中断 — 如果此位为高电平，则信号寄存器位 2 为当前中断源。向此位写入“1”即可将其清空。
- 位 6：信号 1 中断 — 如果此位为高电平，则信号寄存器位 1 为当前中断源。向此位写入“1”即可将其清空。
- 位 5：信号 0 中断 — 如果此位为高电平，则信号寄存器位 0 为当前中断源。向此位写入“1”即可将其清空。
- 位 4：保持寄存器 B 中断 — 如果此位为高电平，则保持寄存器 B 为当前中断源。向此位写入“1”即可将其清空。读取保持寄存器 B 也可将此位清空。
- 位 3：保持寄存器 A 中断 — 如果此位为高电平，则保持寄存器 A 为当前中断源。向此位写入“1”即可将其清空。读取保持寄存器 A 也可将此位清空。
- 位 2：舍入模式 — 指示 DP 处于舍入模式。这意味着 DP 单元输出的任何值都将四舍五入为 16 位的值。
- 位 1：饱和模式 — 指示 DP 单元处于饱和模式。这意味着执行任何所得数字超出 2（24 位）的补数范围的数学与运算都将钳制为模式允许的正数或负数。饱和模式在 DFB 控制器的装配控制中进行设置或取消设置。
- 位 0：RAM 选择 — 显示正在使用的 CS RAM。

Filter_INT_CTRL_REG

此寄存器控制哪些事件生成中断。这些由寄存器位激活的事件一起执行“或”预算，产生 dfb_intr 信号。

位	7	6	5	4	3	2	1	0
值	resvd	resvd	resvd	EN SEM2	EN SEM1	EN SEM0	EN HOLDING REG B	EN HOLDING REG A

如果要使用轮询方法，则可基于所选滤波器通道启用此寄存器的位 0 或 1。此操作在滤波器输出寄存器中数据就绪时生成中断。在状态寄存器中设置相应的状态位。固件可以轮询状态寄存器的相关位，以读取滤波器输出数据。

- 位 7 到位 5：保留
- 位 4：“启用”信号 2 — 如果此位设置为高电平，则每向信号寄存器位 2 写入一个“1”就会产生一次中断。
- 位 3：“启用”信号 1 — 如果此位设置为高电平，则每向信号寄存器位 1 写入一个“1”就会产生一次中断。
- 位 2：“启用”信号 0 — 如果此位设置为高电平，则每向信号寄存器位 0 写入一个“1”就会产生一次中断。
- 位 1：“启用”保持寄存器 B – 如果此位设置为高电平，则每向输出保持寄存器 B 写入一个新有效数据就会产生一次中断。
- 位 0：“启用”保持寄存器 A – 如果此位设置为高电平，则每向输出保持寄存器 A 写入一个新有效数据就会产生一次中断。

PSoC 3 直流和交流电气特性

滤波器直流规范

参数	说明	条件	最小值	典型值	最大值	单位
	DFB 工作电流	F_{DFB} 时为 64 抽头 FIR				
		100 kHz (1.3 ksps)	--	0.03	0.05	mA
		500 kHz (6.7 ksps)	--	0.16	0.27	mA
		1 MHz (13.4 ksps)	--	0.33	0.53	mA
		10 MHz (134 ksps)	--	3.3	5.3	mA
		48 MHz (644 ksps)	--	15.7	25.5	mA
		67 MHz (900 ksps)	--	21.8	35.6	mA

滤波器交流规范

参数	说明	条件	最小值	典型值	最大值	单位
F_{DFB}	DFB 工作频率		DC	--	67	MHz

PSoC 5 直流和交流电气特性

滤波器直流规范

参数	说明	条件	最小值	典型值	最大值	单位
	DFB 工作电流	F_{DFB} 时为 64 抽头 FIR				
		100 kHz (1.3 ksps)	--	0.03	0.075	mA
		500 kHz (6.7 ksps)	--	0.16	0.3	mA
		1 MHz (13.4 ksps)	--	0.33	0.57	mA
		10 MHz (134 ksps)	--	3.3	5.5	mA
		48 MHz (644 ksps)	--	15.7	26	mA
		67 MHz (900 ksps)	--	21.8	35.6	mA

滤波器交流规范

参数	说明	条件	最小值	典型值	最大值	单位
F_{DFB}	DFB 工作频率		DC	--	67	MHz

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
2.0	支持 IIR 滤波器	旧版本中不支持 IIR 滤波器
	重新设计 FIR 滤波器。	旧版本中 FIR 滤波器的窗口版本存在问题。
	滤波器 DMA 向导相关更新	可以使用 DMA 将滤波器设置为数据传输的目标，且无需将 DMA 设置为数据就绪信号。
	在配置窗口中增加显示字段以显示基于滤波器参数设置所需的总线时钟频率	为了解基于参数选择的总线时钟要求。
	滤波器配置窗口更新	为定制系数提供输入选项，同时改进外形和感官效果。
1.50.a	向数据手册添加了特性数据	
	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 向导工具支持滤波器。

© 赛普拉斯半导体公司，2012。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。