2 极低通滤波器数据手册 LPF2 V 2.5

Copyright © 2002-2011 Cypress Semiconductor Corporation. All Rights Reserved.

资源	PSoC® 模块			API 内存 （字节）		引脚 （根据外部 I/O）
	数字	模拟 CT	模拟 SC	闪存	RAM	
CY8C29/27/26/25/24xxx、CY8C23x33、CY8CLED04/08/16、CY8CLED0xD、CY8CLED0xG、CY8CTST120、CY8CTMG120、CY8CTMA120、CY8C28x45、CY8CPLC20、CY8CLED16P01、CY8C28x43、 CY8C28x52						
	0	0	2	120	0	1

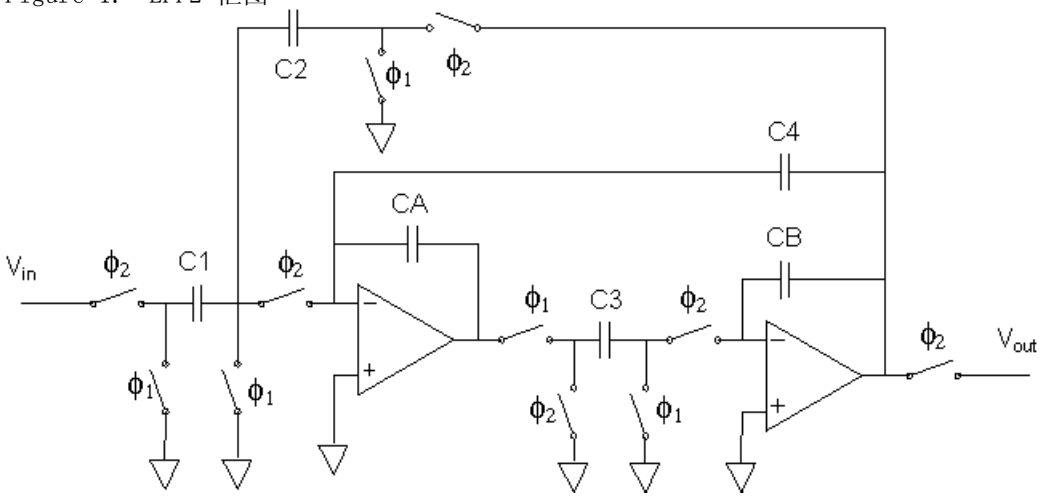
若要获取使用此用户模块的一个或多个完全配置的功能示例项目，请访问 www.cypress.com/psocexampleprojects.

功能和概述

- 用户可编程的增益
- 用户可编程的拐角频率和阻尼比 (damping ratio)，无需外部组件
- 因时钟准确性而直接获得滤波器拐角频稳定性
- 滤波器采样率高达 1.0 MH

LPF2 用户模块实现了一种通用的二阶状态变量滤波器，也称为双二阶低通滤波器。 拐角频率和阻尼比是时钟频率及选定电容值之间比率的函数。 您可以实现任何经典的全极滤波器配置 （巴特沃斯 (Butterworth)、贝塞尔 (Bessel) 和切比雪夫 (Chebyshev)）。 拐角频率可设置到非常精确，也可以通过控制采样率时钟来进行调整。 可通过串联两个或多个 LPF2 用户模块实现四阶函数及更高阶的函数。

Figure 1. LPF2 框图



功能说明

在频率域中，2 极低通滤波器的频率响应如公式 1 中所示：

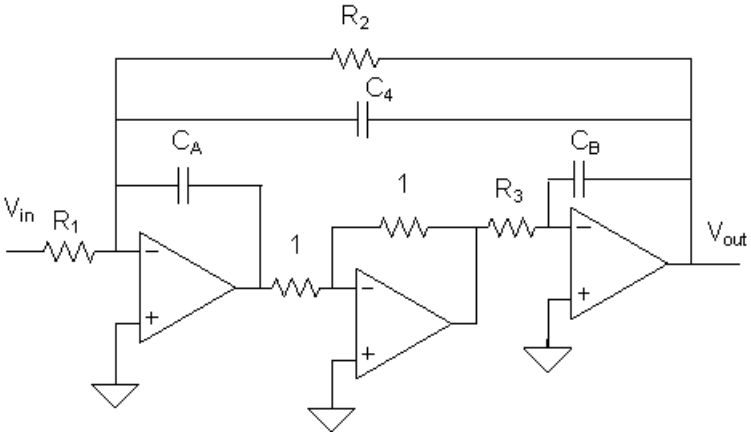
Equation 1

$$\frac{V_{OUT}}{V_{IN}} = \frac{G(\omega_n \omega_0)^2}{s^2 + s\omega_n \omega_0 d + (\omega_n \omega_0)^2}$$

在公式 1 中，d 是阻尼比，ω_0 是固定频率，ω_n 是标准化的 -3 dB 频率。 所有 2 极滤波器均具有渐近于每倍频程 12 dB 的带外衰减 （每极分别为每倍频程 -6 dB）。 带内性能由阻尼比和固定频率确定。 标准巴特沃斯滤波器在带通中具有单调振幅性能和最平坦的相移。 具有低阻尼比的滤波器 （切比雪夫）具有比较平坦的带内振幅特性，但在带通中具有非线性相移并且脉冲响应表现出振荡的特性。 具有高阻尼比的滤波器 （贝塞尔）在带通中具有线性相移并具有过冲最低的脉冲响应特性，但其近带外衰减较低。 d 和 ω_0 的值可以很容易地在任何滤波器设计参考中找到。 通过调整开关电容 PSoC 模块中电容之间的比率，可实现所有这些滤波器形式。

双二阶滤波器的基本形式是一对积分器，其中包含受控直流通路和与频率相关的反馈通路。 若要了解双二阶滤波器，请查看图 2 中显示的标准 RC 形式。

Figure 2. RC 双二阶滤波器框图



典型的 RC 双二阶低通滤波器采用 3 个运算放大器。 RC 双二阶滤波器的传输函数如公式 2 中所示：

Equation 2

$$\frac{V_{OUT}}{V_{IN}} = \frac{-\frac{R_2}{R_1}\frac{1}{R_2 R_3 C_A C_B}}{s^2 + s\frac{C_4}{R_3 C_A C_B} + \frac{1}{R_2 R_3 C_A C_B}}$$

在 PSoC 开关电容实现中，通过反向输出模块增益的极性，省去了中央反相运算放大器。 通过比较 RC 双二阶滤波器和开关电容实现的示意图，可以看到电阻换成了开关电容。

[+] Feedback

由于开关电容电路本质上是作为时间采样器件，因此将在时间域中开发传输函数，其中 z^{-1} 是一个采样周期的时间延迟，而不是频率域（s=jω）。 使用双线性变换将传输函数转换为频率域。 传输函数将解析为公式 3：

Equation 3

$$\frac{V_{OUT}}{V_{IN}} = \frac{-\frac{C_1}{C_2}\frac{\left(1-\left(\frac{s}{2f_s}\right)^2\right)f_s^{\,2}}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}}{s^2+\frac{C_4}{C_2}\frac{sf_s}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}+\frac{f_s^{\,2}}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}}$$

使用公式 1 的标准形式计算公式 3 会产生一组用于增益 G，拐角频率 ωnω0 和阻尼比 d 的设计公式：

Equation 4

$$G = -\frac{C_1}{C_2}$$

Equation 5

$$\omega_n\omega_0 = \frac{f_s(C_2C_3)^{\frac{1}{2}}}{\left(C_AC_B-\frac{C_2C_3}{4}-\frac{C_4C_3}{2}\right)^{\frac{1}{2}}}$$

Equation 6

$$d = \frac{C_4\left(\frac{C_3}{C_2}\right)^{\frac{1}{2}}}{\left(C_AC_B-\frac{C_2C_3}{4}-\frac{C_4C_3}{2}\right)^{\frac{1}{2}}}$$

公式 3 的分子包含以下项， $1-(s/(2f_s))^2$ 该项会导致在信号频率接近奈奎斯特率一半 （即采样率 fS）时滤波器衰减降低。 采样率越高，滤波器的性能就更接近于公式 1 的标准形式，并且波形也会更平滑。

针对各种应用的功耗设置

开关电容模块功耗和偏压设置决定了滤波器的性能。 使用用户模块 API 选择功耗模式。 使用 PSoC Designer 的 “ 全局资源 ” 窗口中的 “ 运算放大器偏压 ” 参数选择偏压。 功耗和偏压设置决定了运算放大器工作电流，而后者又决定了斜率 (SR)。 当功耗和偏压均设置为高时，运算放大器的斜率为 4 V/μsec。 功耗和偏压每降低一级，斜率就会降低 ½。

[+] Feedback

提供逼真信号所需的斜率取决于峰值电压（V_{pk}）和频率（F）。 在一个时钟相位的一半所经过的时间内，滤波器的输出信号会从其旧值上升到新值。 因此，滤波器要求运算放大器斜率为信号斜率的四倍。

Equation 7

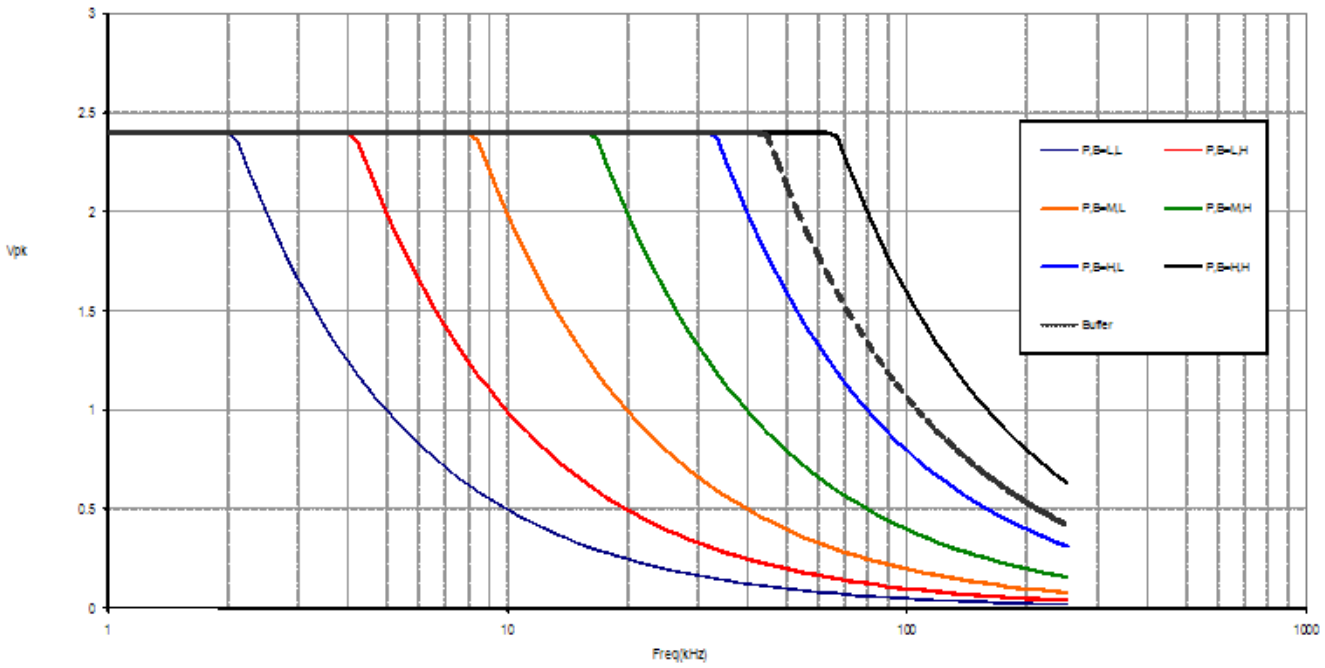
$$SR = 4 \times (2\pi V_{pk} F)$$

下表中给出了 Vdd = 5.0 V 的轨至轨输出的最大工作频率：

功耗，偏压	kHz
P,B=H,H	64
P,B=H,L	32
P,B=M,H	16
P,B=M,L	8
P,B=L,H	4
P,B=L,L	2

在此表中，P = 功耗，B = 运算放大器偏压，L = 低，H = 高，M = 中。 对于较低的信号电平，允许的峰值电压如下图所示：

Figure 3. 较低信号电平的峰值电压



滤波器输出是内部信号。 如果滤波器在芯片外驱动，则必须进行缓冲。 缓冲区的斜率为 0.65 V/µsec。 缓冲区的工作频率限制如图 3 所示。

[+] Feedback

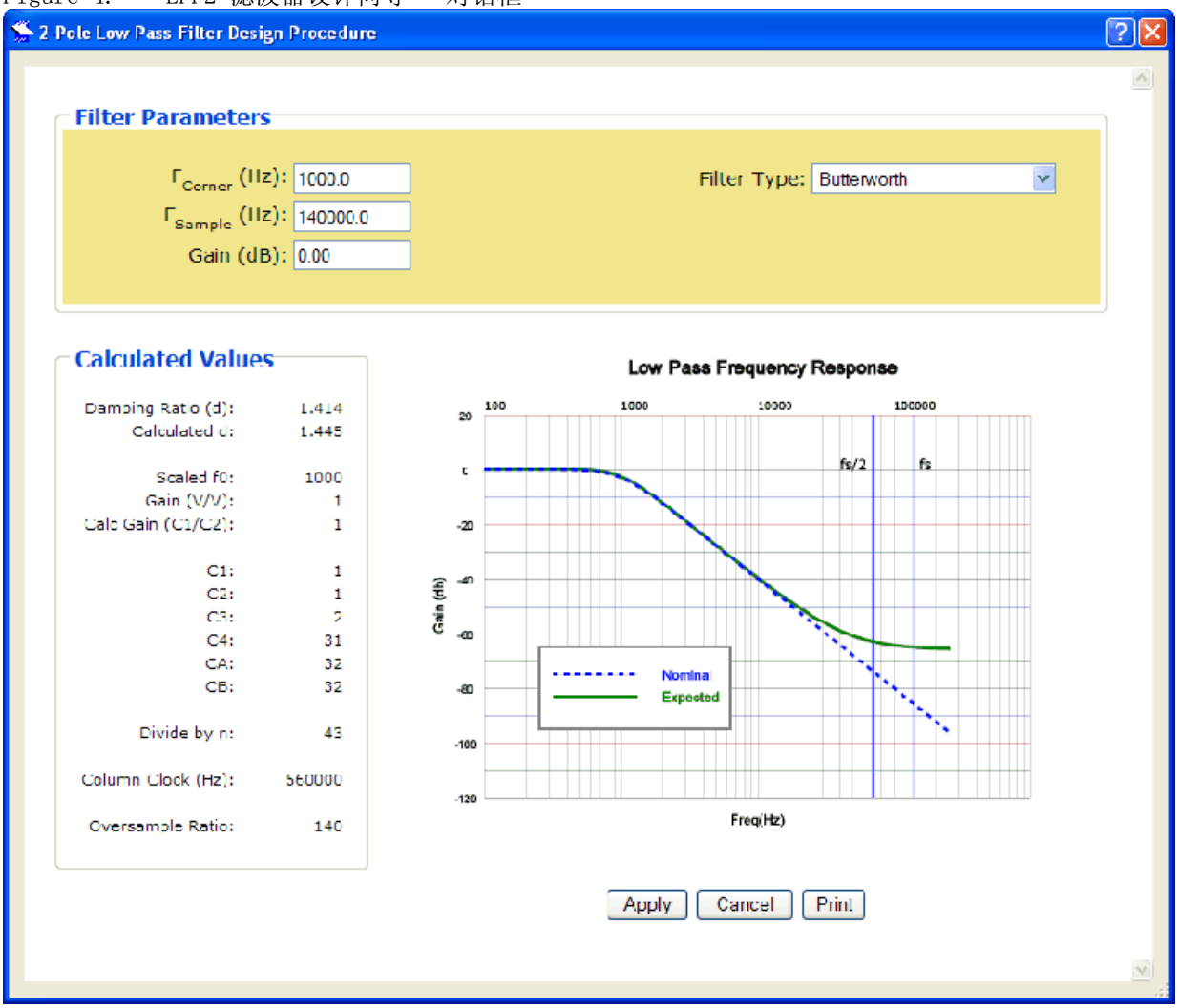
滤波器设计

设计目标通常是实现最高可能采样率（f_{clk}），以获得最佳波形逼真度并最大限度地减少带外信号的锯齿。其他系统要求（例如，共享时钟）可决定采样率；可以定制电容器值以获得所需的采样率。

LPF2 为确定电容器值提供了三种备选方法。 PSoC Designer 提供了滤波器设计向导，可自动完成 2 极滤波器的设计过程。 也可在电子表格 *LPF2 Design.xls* 中实现这一过程。 您可以从 PSoC Designer 帮助菜单中的 “ 文档 ...” 菜单项访问此电子表格。 在单独的 Microsoft Excel 电子表格 *LPF4 Design.xls* 中自动完成针对 2 极对（四阶）的类似设计过程。 可在电子表格中实验性地修改由该向导施加的设计约束。 若要尽可能地手动控制设计过程，请参见本文末尾附录中可手动执行的计算过程。 此外，还以拐角频率为 1 kHz 的巴特沃思滤波器为例，说明了该过程的工作原理。

若要使用 PSoC Designer 的内置 “ 滤波器设计向导 ”，请先将 LPF2 实例放置在模拟阵列中。 右键单击该用户模块，然后从弹出菜单中选择 “ 滤波器设计向导 ...”。 此时会出现一个对话框（如图 3 所示），其中介绍了设计传输函数的简单重复过程。

Figure 4. “LPF2 滤波器设计向导 ” 对话框



在该对话框向下滚动可显示用于绘制振幅响应的值表。 可将该表中的值剪切并粘贴到电子表格或其他工具中，以进行进一步图解和分析。

[+] Feedback

直流和交流电气特性

The following values are indicative of expected performance and based on initial characterization data. 除非另行指定，否则在以下条件下保证所有限值：T_J =+25C，V_{dd} = 5.0V，功耗为高，运算放大器偏压为低，以模拟接地为参考的输出电压 = 2*$V_{BandGap}$。

Table 1. 5.0V LPF2 直流电气特性

参数	典型值	限制	单位	条件和注释
直流偏移电压 [1]	18	-.	mV	以模拟接地为参考
直流增益误差 [2]	1.55	-.	.	
运行电流				
低功耗	290	-.	µA	
中功耗	1065	-.	µA	
高功耗	4015	-.	µA	

Table 2. 5.0V LPF2 交流电气特性

参数	典型值	限制	单位	条件和注释
最大时钟频率 [3]				
低功耗	-.	2	MHz	
中功耗	-.	4	MHz	
高功耗	-.	8	MHz	
拐角频率误差	2.3	-.	.	与额定值之间的偏差 [2]
阻尼比误差	1.85	-.	.	
噪声 [4]	2140	.	nV/√Hz	

The following values are indicative of expected performance and based on initial characterization data. 除非下表中另行指定，否则在以下条件下保证限值：TA = 25C，V_{dd} = 3.3 V，功耗为高，运算放大器偏压为低，以模拟接地为参考的输出电压 = V_{dd}/2。

Table 3. 3.3V LPF2 直流电气特性

参数	典型值	限制	单位	条件和注释
直流偏移电压 [1]	14	-.	mV	以模拟接地为参考
直流增益误差 [2]	1.65	-.	.	

[+] Feedback

参数	典型值	限制	单位	条件和注释
运行电流				
低功耗	225	-.	µA	
中功耗	850	-.	µA	
高功耗	3275	-.	µA	

Table 4. 3.3V LPF2 交流电气特性

参数	典型值	限制	单位	条件和注释
最大时钟频率 [3]				
低功耗	-.	2	MHz	
中功耗	-.	4	MHz	
高功耗	-.	8	MHz	
拐角频率误差	2.15	-.	.	与额定值之间的偏差 [2]
阻尼比误差	1.9	-.	.	
噪声 [4]	1880	-.	nV/√Hz	

电气特性注释

1. 使用具有以下参数的 1 kHz 滤波器计算出的典型直流偏移电压：Qs 为 3、5 和 15； C2 = 1 至 16 ； C3 = 3、10 和 25。 使用滤波器设计电子表格计算出 C1 和 C4。
2. 从具有以下参数的额定滤波器确定的偏差值：fcenter=1 kHz （巴特沃思），单位增益，C1=1，C2=3，C3=31，C4=1，fclock=20.3 kHz，Q=10。
3. 采样率是列时钟频率的四分之一。
4. 使用 10 kHz 滤波器时在 1 kHz 处计算出的噪声。

除非另行说明，否则在以下条件下保证所有限值：T_J = -40C 至 +85C，V_{dd} = 5.0V ±10%，功耗为高，运算放大器偏压为低，以模拟接地为参考的输出电压 = 2*$V_{BandGap}$。

Table 5. 5.0V LPF2 直流电气特性

参数	典型值 [1]	限值 [2]	单位	条件 [1] 和注释
偏移电压 [5]	42		mV	请参见图
直流增益误差	1.55		.	参考设计 = 1.0 kHz 巴特沃思 [3]
运行电流				
低功耗	250	-.	µA	
中功耗	560	-.	µA	
高功耗	1560	2000	µA	

[+] Feedback

Table 6. 5.0V LPF2 交流电气特性

参数	典型值[1]	限值[2]	单位	条件和注释
拐角频率偏差	-.	2.5	.	与额定值之间的偏差
阻尼比偏差	.	2.6	.	与额定值之间的偏差
噪声	-.	8.0	mV rms	
时钟频率				
低功耗	-.	2.0	MHz	
中功耗	-.	4.0	MHz	
高功耗	-.	8.0	MHz	

除非下表中另行说明，否则在以下条件下保证限值：TA = -40C 至 +85C，V_{dd} = 3.0 至 3.6 V，功耗为高，运算放大器偏压为低，以模拟接地为参考的输出电压 = $V_{dd}/2$。

Table 7. 3.3V LPF2 直流电气特性

参数	典型值[1]	限值[2]	单位	条件和注释
偏移电压[5]	42		mV	请参见图
直流增益误差	1.65		.	参考设计 = 1.0 kHz 巴特沃思[3]
运行电流				
低功耗	200	-.	µA	
中功耗	500	-.	µA	
高功耗	1280	1800	µA	

Table 8. 3.3V LPF2 交流电气特性

参数	典型值[1]	限值[2]	单位	条件和注释
拐角频率偏差[3]	-.	2.5	.	
阻尼比偏差[3]	-.	2.6	.	
噪声	-.	6.0	mV rms	

[+] Feedback

参数	典型值 [1]	限值 [2]	单位	条件和注释
时钟频率 [4]				
低功耗	-.	1.0	MHz	
中功耗	-.	2.0	MHz	
高功耗	-.	4.0	MHz	

电气特性注释

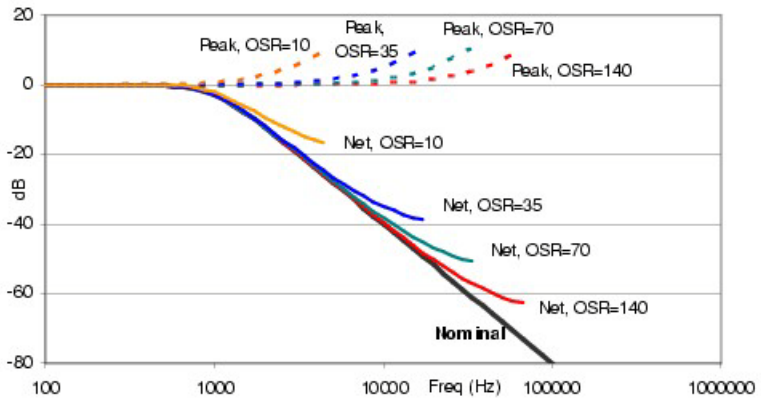
1. 典型值表示 +25C 时的参数标准。
2. 限值将通过测试或统计分析加以保证。
3. 参考设计 C1=1，C2=1，C3=2，C4=31，CA=32，CB=32，f(sample)=139.2 kHz，阻尼比 = 1.392。
4. 时钟频率适用于注释 3 的滤波器，使用时钟调节拐角频率。
5. 1kHz 滤波器，fclk=35 kHz，水平反相拓扑结构，C1=C2=8，C3=16，C4=31，CA=CB=32。

性能说明

滤波器增益和相位的误差只有在拐角频率附近才比较明显。滤波器电容器值可以采用其他选项。可以调节 C2 与 C3 之比，以允许阻尼比有更好的分辨率。 将 CA 或 CB 设置为 16 可在阻尼比设置中增加分辨率，但代价是减少了取决于最大可能值的采样率。

因采样率增加而产生的信号保真度方面的有利因素会超出因极位置不理想而产生的函数误差方面的不利因素。 降低采样率 （通过增加 C1 和 C2）具有降低接近尼奎斯特率的衰减的效果。

Figure 5. 滤波器性能与过采样率



[+] Feedback

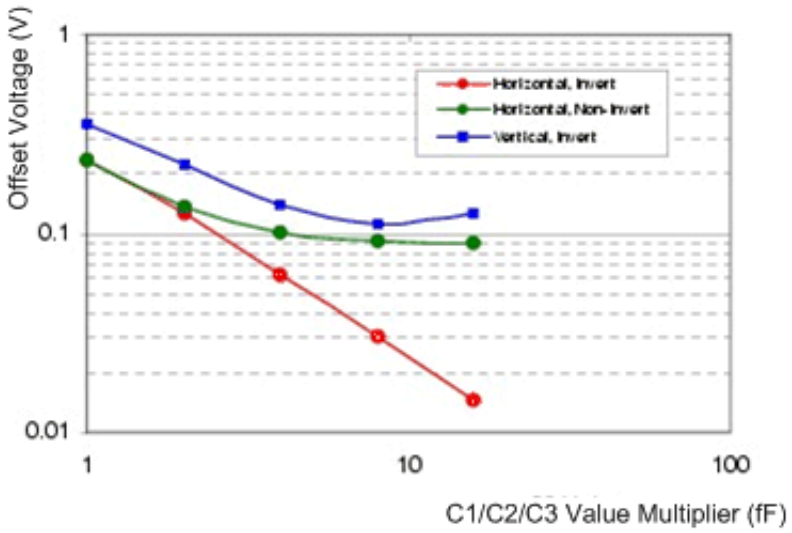
由于运算放大器开环增益、绝对电容值和滤波器放置方式的影响，滤波器性能在一定程度上与标准形式有所偏离。 虽然高于 100 kHz 的低通滤波器在数学上是可行的，但运算放大器的性能限制会使这些选择变得不可靠。 在 “ 全局参数 ” 窗口中，应将拐角频率高于 40 kHz 的滤波器的用户模块功耗设置为 “ 高功耗 ”，将运算放大器偏压设置为 “ 高 ”。 对于给定的滤波器特性，可能有许多电容值解决方案满足这一要求，但每个变体都具有不同的采样率。 通常，增加电容值会降低过采样率 （即采样率与额定 -3dB 频率之比）。 为了获得最高过采样率以及最平滑的输出波形，应在设计电子表格、设计向导中或通过使用设计公式 “ 手动 ” 进行推导，从 C2 的最小可能值开始构建滤波器。 例如，C2=1、C4=31 的巴特沃思滤波器具有 140 的过采样率。 设置 C2=2 将获得 70 的过采样率。 在所有情况下，选定的采样率必须小于最高频率 1.00 MHz （列时钟 = 4.0 MHz）；为满足此要求，较高频率的滤波器将需要采用较低的过采样率。 公式 3 的分子显示了一对处于 2fs 处的一对复零点。 随着采样频率的降低，这些零点的作用会不断地增强，从而导致带上边沿出现峰值，如下图所示。

内部开关拓扑结构中的电荷注入限制了滤波器的直流性能。 这种影响存在一种随机部分和一种由放置方式决定的部分。

Note 放置在水平模块对中的滤波器的直流偏移低于放置在垂直模块对中的滤波器。 通过增加 C2 （以及 C1 和 C3）的值来调节滤波器，从而减少直流偏移误差。 有关 C2 值对偏移误差有何影响的图形表示形式，请参见下图。

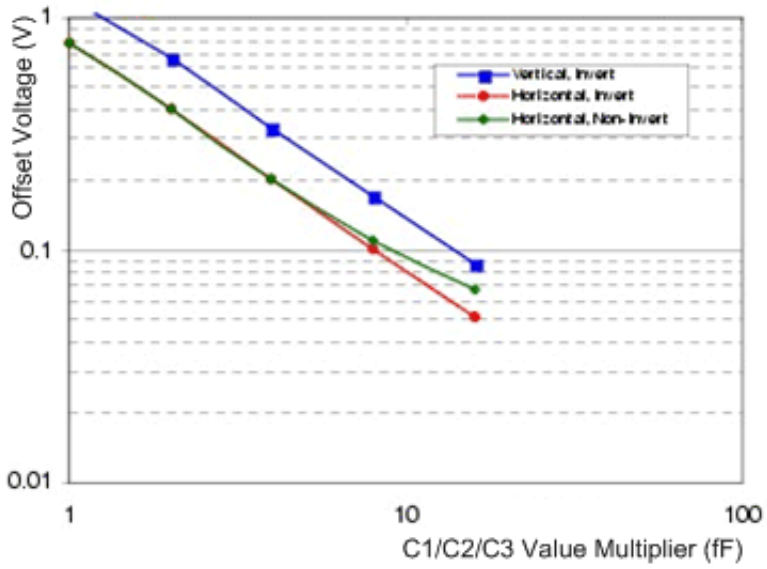
C1、C2 和 C3 值是输入 FPF2 向导中的复用器。 这些复用器是器件数据手册中直流模拟 PSoC 模块规范中的 Csc 的系数。

Figure 6. 取决于电容值 (C1/C2/C3) 的典型偏移，适用于 PSoC 器件 （不包括 CY8C26xxx/25xxx）



[+] Feedback

Figure 7. 取决于电容值 (C1/C2/C3) 的典型偏移，适用于 CY8C25xxx/26xxx



放置

器件编辑器将逻辑模块 FLIN 和 FLOUT 映射到器件模拟阵列中的一对相邻的开关电容 PSoC 模块上。 可使用多种方法从模拟 PSoC 模块构建双二阶滤波器电路。 每种构建方法都实现了 LPF2 框图，但在 FLIN 和 FLOUT 模块中所使用的电容和连接各不相同。 每种方法所生成的电路拓扑结构各不相同，并且其中的映射和 I/O 结果也各不相同。 最明显的区别就是这两个 PSoC 模块在模拟阵列内是排成一行还是排成一列。 这些拓扑结构还决定了在阵列中可与其他模块建立哪些连接。 但无论选择的拓扑结构如何，滤波器输入始终连接至 FLIN 模块并且输出始终由 FLOUT 模块驱动。

每次创建 LPF2 用户模块的实例时，PSoC Designer 都会提供一个对话框，其中包含可帮助选择电路拓扑结构的插图和文本。 可以随时变更选项，方法是：在选择栏中右键单击用户模块图标；如果已放置用户模块，则右键单击它的某一 PSoC 模块，然后从弹出菜单中选择 “ 选择用户模块选项 ...”。 如果在放置后更改拓扑结构，则需要重新将用户模块放入模拟阵列中。

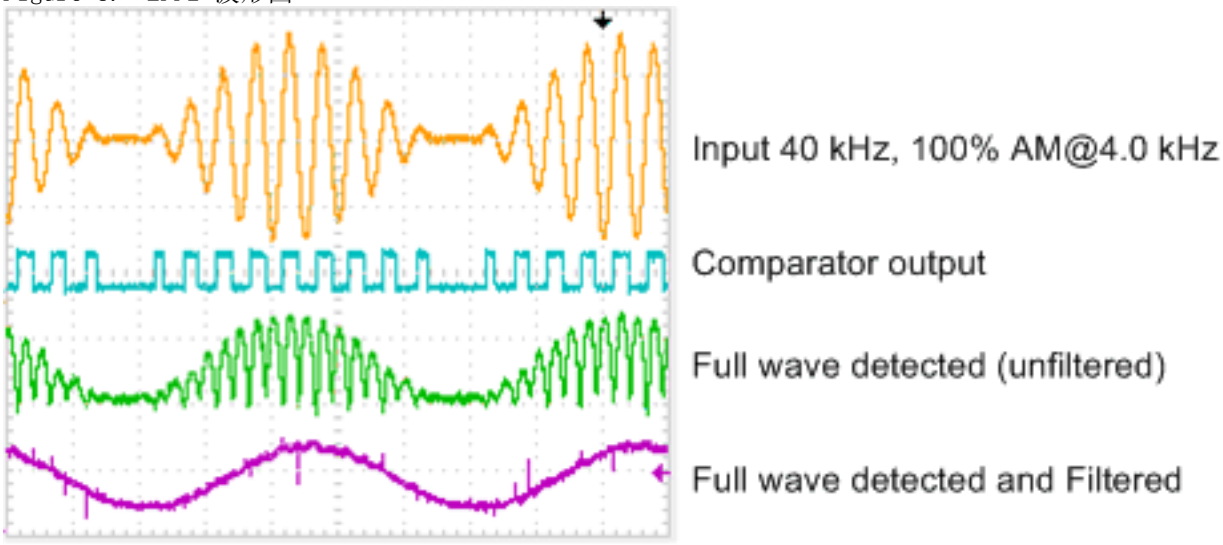
极性和调制器功能

输入开关电容模块具有极性控制参数。 也可以通过调制器所包含模块之外的时钟来控制极性。 调制器用于根据应用要求上移或下移频率，或与比较器组合在一起以构成完整的波形检测器。 调制函数在滤波率或比较器输出极性处将输入乘以 +1 或 -1。 这会在载波与输入频率的叠加处以及载波与输入频率的差分处生成信号。 调制载波应具有 50% 的负载循环，以最大限度地减少偶次谐波锯齿。

该示例显示了在 4.0 kHz 处经过 100% AM 调制的 40 kHz 输入信号。 该输入馈送给比较器以及 LPF2 的输入以提供零交叉功能。 比较器的输出选定为调制器输入。 在比较器波形下方显示了检测信号的完整波形形式 （用单独模块生成，仅供显示）。 低通滤波器的输出显示在底部跟踪中。 检测信号的峰至峰电平比原始调制信号小 30%，这是因为矫正后的完整正弦波形的平均电平是峰值电平的 sqrt(0.5) 倍。

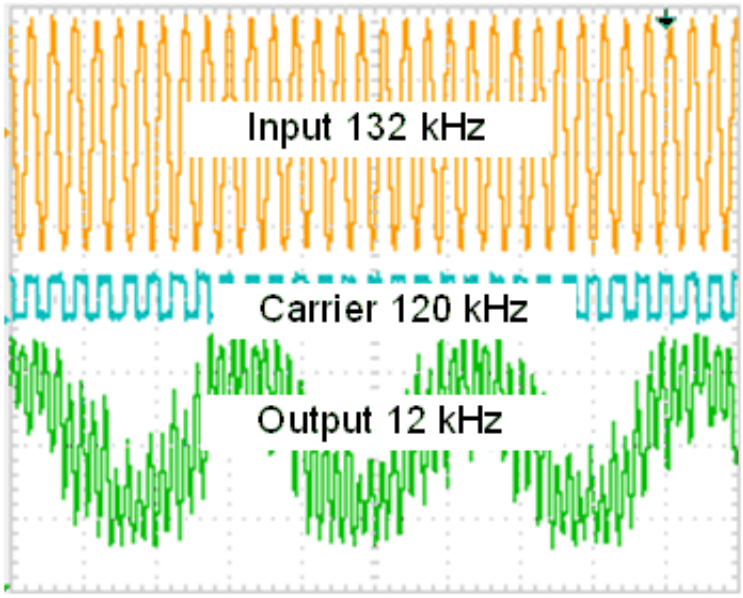
[+] Feedback

Figure 8. LPF2 波形图

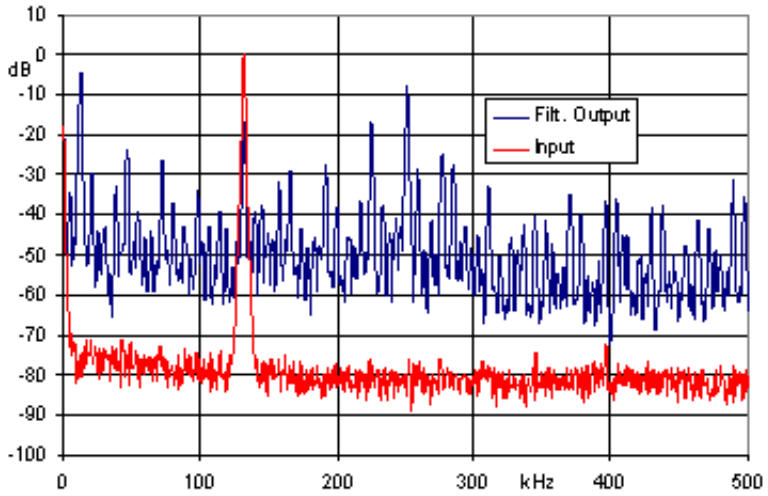


下移示例使用由 120 kHz 方波调制的 132 kHz 信号生成 12 kHz 差分信号。 在 252 kHz 处生成了一个同等大小的信号，但此信号大部分将被低通滤波器滤出。 下面显示了未滤出的下行调制信号示例。

Figure 9. 未滤出的下行调制信号

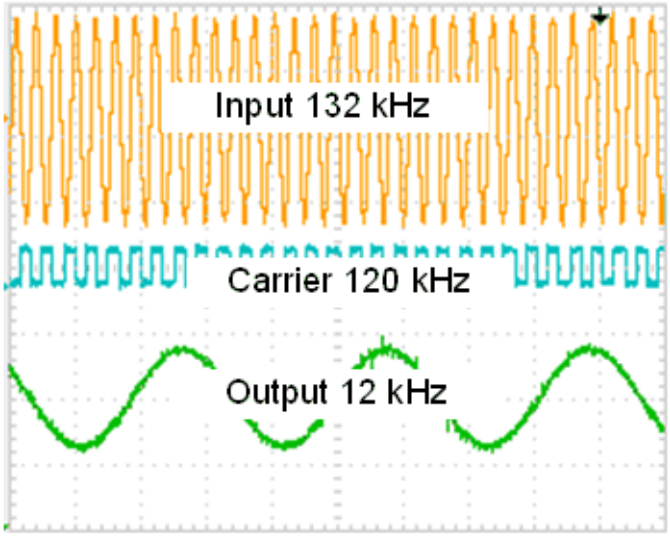


[+] Feedback

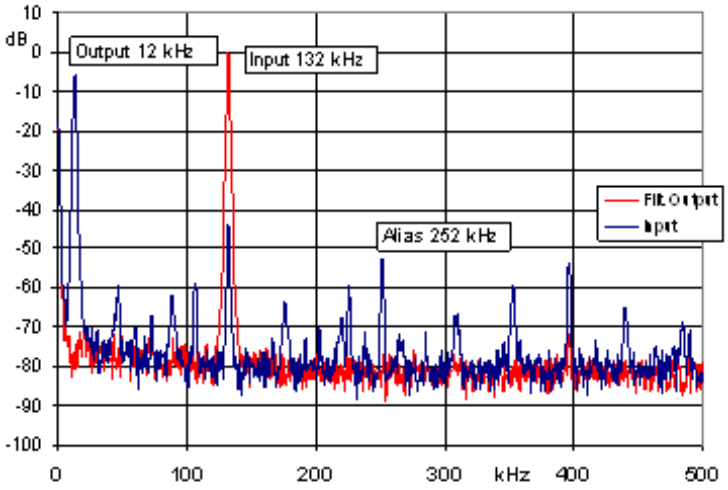


配置有 14 kHz 截止频率的 LPF2 显著地减少了谐波和锯齿，从而产生了以下波形。 调制和滤波均在同一模块中完成。 可使用 LPF4 用户模块实现进一步减少锯齿和残余输入的目的。

Figure 10. LPF2 波形



[+] Feedback



通过选择适当的全局输出或广播总线，或选择比较器总线输出以构建完整的波形检测器，可以从多个数字模块源中的任何一个选择调制器信号。

参数和资源

若要构建低通滤波器，请将 LPF2 用户模块的一个实例放置在器件编辑器的模拟阵列中。 使用设计过程选项确定该滤波器的电容值，然后连接输入并配置模拟总线连接和时钟资源。 这些参数将在本节中进行说明。

输入

滤波器的输入由相邻 PSoC 模块的输出来驱动。 可以在器件编辑器中选择输入。

AnalogBus

用户模块的 FLOUT 模块的输出可连接至相邻的 PSoC 模块。 在器件编辑器中建立从其他用户模块至此输出的连接。 可使用 AnalogOutBus_x 选项（其中 “x” 是列号）将 FLOUT 模块的输出连接至模拟列输出总线。 这可以连接至同一列的模拟输出缓冲区，并防止模拟输出总线访问同一列中的其他用户模块。 所有互连均使用器件编辑器进行配置。

CompBus

FLOUT 模块比较器输出可路由至数字 PSoC 模块的输入总线，也可路由至中断。 若要进行其中任何连接，必须将 CompBus 参数设置为 “ 启用 ”。

电容值 C1、C2、C3、C4、CA 和 CB

这六个电容值之间的比率决定了滤波器的频率和相位响应。 这些名称指 LPF2 框图中所绘制的电容。 CA 和 CB 均可采用 16 或 32 个单位电容值。 C1 到 C4 采用的值为 0 到 31 （但要使传输功能有效，值必须大于零）。 可通过自动或手动执行相应步骤完成传输功能的设计。 若要访问内置设计工具，请右键单击放置的滤波器，然后从弹出菜单中选择 “ 滤波器设计向导 ...”。 有关设计的详细信息，请参见 “ 滤波器设计 ” 部分。

极性 （仅限 A 输入拓扑结构滤波器）

此参数确定输出极性 （相对于输入极性）。 可通过选择*反相*将输出设置为与输入信号反相，也可以选择*同相*使输入和输出保持相同的极性。 此参数仅适用于 A 输入拓扑结构滤波器。

[+] Feedback

采样时钟

低通滤波器所需的采样时钟将使用 “ 功能说明 ” 一节中的设计公式计算得出。 与前面列出的其他用户模块参数不同，采样时钟不会显示在器件编辑器的全局资源列表下的用户模块参数列表中。 与特定于特殊用户模块的信号输入不同，每个采样时钟均为整个模拟列提供服务。 两个 PSoC 模块的采样率时钟必须相同，当选择水平放置模块后，两个列时钟必须由同一时钟源来驱动。 每个列时钟发生器均将其输入四分频，以在模块中生成 Φ1 和 Φ2 内部时钟，因此时钟源的速度必须比所需的滤波器采样时钟快四倍。

可选择的时钟源包括任何数字 PSoC 模块和系统时钟分频器。 当系统时钟分频器必须指派用于其他用途，所有定时器、计数器和脉冲宽度调制器 (PWM) 用户模块均是适合的选择。

在器件编辑器中使用每列的 CLK 复用器选择列时钟的时钟源。 系统时钟是此复用器的直接输入。 如果使用 PSoC 模块生成时钟，则通过 ACLK0 和 ACLK1 复用器将这些模块连接到 CLK 复用器。

调制器时钟

调制器使用输入模块 (FLIN) 的 A 输入中的极性控制功能。 此函数可用于 LPF2A 和 LPF2VA 拓扑结构。 它不能用于 LPF2B 和 LPF2V，因为这些拓扑结构使用输入模块的 B 输入，该输入没有极性控制功能。

提供了八个时钟源 （包括关闭状态）：

Table 9. 调制器时钟源选择

参数	注释
无	调制器时钟源处于关闭状态
GlobalOutEven_0	从 GlobalOutOdd_0 获取调制器时钟
GlobalOutEven_1	从 GlobalOutOdd_1 获取调制器时钟
Row_0_Broadcast	从 Row_0_Broadcast 获取调制器时钟
ComparatorBus_0	从 ComparatorBus_0 获取调制器时钟
ComparatorBus_1	从 ComparatorBus_1 获取调制器时钟
ComparatorBus_2	从 ComparatorBus_2 获取调制器时钟
ComparatorBus_3	从 ComparatorBus_3 获取调制器时钟

完整的波形检测器通常是采用以下方式构建的：使用 PGA 或 BPF 的输出驱动 LPF 的信号输入，并使用比较器的输出和 BPF 的比较器输出驱动 LPF 的调制器输入。

应用程序编程接口

应用程序编程接口 (API) 子程序作为用户模块的一部分提供，使您可以以更高的水平处理模块。 本节指定每个函数的接口，以及 “ 包含 ” 文件所提供的相关常量。

Note 在这里，如同所有用户模块 API 中的一样，A 和 X 寄存器的值可能通过调用 API 函数发生更改。 如果在调用后需要 A 和 X 的值，则调用函数负责在调用前保留 A 和 X 的值。 此 “ 寄存器易失 ” 策略是针对提高效率的目的选择，自 PSoC Designer 1.0 版起已强制使用此策略。 C 编译器自动遵循此要求。 汇编语言程序员也必须确保其代码遵守这一策略。 虽然一些用户模块 API 函数可以保留 A 和 X 不变，但是无法保证它们将来也会如此。

所提供的入口点用于初始化 LPF2 用户模块、更改电源设置以及禁用该用户模块。

[+] Feedback

LPF2_Start

说明：

为此用户模块执行所有必需的初始化操作，并为开关电容 PSoC 模块设置功耗等级。

C 原型：

```
void  LPF2_Start(BYTE bPowerSetting)
```

汇编：

```
mov   A, bPowerSetting
lcall  LPF2_Start
```

参数：

bPowerSetting： 指定两个模拟 PSoC 模块的功耗等级的一个字节。 在复位和配置后，分配给仪表放大器的 PSoC 模块将降低功耗运行。 下表列出了以 C 语言和汇编语言提供的符号名称及其相关值：

符号名称	值
LPF2_OFF	0
LPF2_LOWPOWER	1
LPF2_MEDPOWER	2
LPF2_HIGHPOWER	3

Note 为了获得正常性能，拐角频率高于 40 kHz 的滤波器应 (1) 使用 LPF2_HIGHPOWER，(2) 在 “ 全局参数 ” 窗口中将全局参数 “ 运算放大器偏压 ” 设置为 “ 高 ” 。

返回值：

无

副作用：

此函数可能更改 A 和 X 寄存器。

LPF2_SetPower

说明：

为开关电容 PSoC 模块设置功耗等级。 可用于关闭和打开用户模块中的模块。

C 原型：

```
void  LPF2_SetPower(BYTE bPowerSetting)
```

汇编：

```
mov   A, bPowerSetting
lcall  LPF2_SetPower
```

参数：

bPowerSetting： 与 Start 入口点所采用的 bPowerSetting 相同。

返回值：

无

[+] Feedback

副作用：

此函数可能更改 A 和 X 寄存器。

LPF2_SetCA、SetCB

说明：

为用户模块 FLIN 模块 (CA) 和 FLOUT 模块 (CB) 中的反馈电容设置值。 这可以随时修改低通滤波器的传输功能。

C 原型：

```
void  LPF2_SetCA(BYTE FEEDBACK_CONSTANT)
void  LPF2_SetCB(BYTE FEEDBACK_CONSTANT)
```

汇编：

```
mov   A, FEEDBACK_CONSTANT
lcall  LPF2_SetCA                ; or, call  LPF2_SetCB
```

参数：

FEEDBACK_CONSTANT: 一个指定反馈电容 CA 或 CB 大小的字节（请参见 LPF2 框图）。 在 C 语言和汇编语言 include 文件中提供符号名称；下表列出了这些符号名称的相关值：

符号名称	值
LPF2_FEEDBACK_16	0x00
LPF2_FEEDBACK_32	0x01

返回值：

无

副作用：

此函数可能更改 A 和 X 寄存器。

LPF2_SetC1、SetC2、SetC3 和 SetC4

说明：

为用户模块中的特定电容设置值。 这可以通过修改 C1 来调整增益，通过调整其他值来改变滤波器的传输特性。

C 原型：

```
void  LPF2_SetC1(BYTE bCapValue)
void  LPF2_SetC2(BYTE bCapValue)
void  LPF2_SetC3(BYTE bCapValue)
void  LPF2_SetC4(BYTE bCapValue)
```

汇编：

```
mov   A, bCapValue
lcall  LPF2_SetC1                ; or, call  LPF2_SetC2 (or SetC3 or SetC4)
```


[+] Feedback

参数：

CapValue： C1、C2、C3 和 C4 的整数值 1 到 31 （请参见 LPF2 框图）。 超出此范围以外的值将截断为 32 的模数。

返回值：

无

副作用：

此函数可能更改 A 和 X 寄存器。

LPF2_SetPolarity （仅限 A 输入拓扑结构滤波器）

说明：

通过选择是否对 FLIN 的输入信号进行反相，设置输出信号的极性。 这可以随时修改低通滤波器的输出极性。 此函数仅适用于 A 输入拓扑结构滤波器。

C 原型：

```
void  LPF2_SetPolarity(BYTE FEEDBACK_CONSTANT)
```

汇编：

```
mov   A, FEEDBACK_CONSTANT
lcall  LPF2_SetPolarity
```

参数：

POLARITY_CONSTANT: 一个指定是否反相的字节。 在 C 语言和汇编语言 include 文件中提供符号名称；下表列出了这些符号名称的相关值：

符号名称	值
LPF2_POLARITY_INVERTING	0x00
LPF2_POLARITY_NON_INVERTING	0x01

返回值：

无

副作用：

此函数可能更改 A 和 X 寄存器。

LPF2_Stop

说明：

关闭用户模块的电源。

C 原型：

```
void  LPF2_Stop(void)
```

汇编：

```
lcall  LPF2_Stop
```


[+] Feedback

参数：

无

返回值：

无

副作用：

此函数可能更改 A 和 X 寄存器。

固件源代码示例

在 C 语言中，使用低通滤波器十分简单，就像使用 Start API 开始操作以及在完成时调用 Stop API 一样。

```
#include "LPF2.h"

LPF2_Start(LPF2_HIGHPOWER);
...                                 // (application processing)
LPF2_Stop();
```

可以随时设置和修改滤波器的传输功能。 假定模拟列时钟由分频器为 304 的 24 MHz 时钟源来驱动，则以下代码创建一个中心频率为 1000 Hz、Q 为 10 的滤波器，然后启动该滤波器。 请注意，无论选择哪一种拓扑结构或者其在模拟 PSoC 模块阵列中的放置位置如何，都将使用这一相同的代码。

```
LPF2_SetC1(  1 );
LPF2_SetC2( 10 );
LPF2_SetC3( 10 );
LPF2_SetC4(  3 );
LPF2_SetCA( LPF2_FEEDBACK_32 );
LPF2_SetCB( LPF2_FEEDBACK_32 );
LPF2_SetPolarity( LPF2_POLARITY_INVERTING );
LPF2_Start(LPF2_HIGHPOWER);
```

汇编语言等效代码如下：

```
include "LPF2.inc"

mov    A, 1
lcall  LPF2_SetC1
mov    A, 10
lcall  LPF2_SetC2
mov    A, 10
lcall  LPF2_SetC3
mov    A, 3
lcall  LPF2_SetC4
mov    A, LPF2_FEEDBACK_32
lcall  LPF2_SetCA
mov    A, LPF2_FEEDBACK_32
lcall  LPF2_SetCB
mov    A, LPF2_POLARITY_INVERTING
lcall  LPF2_SetPolarity
mov    A, LPF2_HIGHPOWER
lcall  LPF2_Start
```


[+] Feedback

Note 设计公式显示增益与 C1 的值成比例，但拐角频率和阻尼与该值无关。 选择传输功能后，可使用 LPF2_SetC1 API 函数实现可编程增益控制。

配置寄存器

LPF2 用户模块的拓扑结构和放置位置决定了所用模拟开关电容 PSoC 模块的配置寄存器中过半位数的内容。 其中，与放置位置无关的那些位由寄存器表中的固定值表示。 在可变位字段中，大多数是通过选择输入和传输功能设计确定的。 本节末尾的寄存器定义中所用可变位字段的定义如下。

水平 A 输入拓扑结构

Table 10. 模块 FLIN: 寄存器 CR0

位	7	6	5	4	3	2	1	0
CR0	CA	0	极性	C1				
CR1	输入			C2				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	反馈		电源	

Table 11. 模块 FLOUT: 寄存器 CR0

位	7	6	5	4	3	2	1	0
CR0	CB	0	0	C3				
CR1	FBIN			0	0	0	0	0
CR2	AnalogBus	CompBus	0	C4				
CR3	0	0	1	0	0	1	电源	

[+] Feedback

水平 B 输入拓扑结构

Table 12. 模块 FLIN

位	7	6	5	4	3	2	1	0
CR0	CA	0	1	C2				
CR1	反馈			C1				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	输入		电源	

Table 13. 模块 FLOUT

位	7	6	5	4	3	2	1	0
CR0	CB	0	0	C3				
CR1	FBIN			0	0	0	0	1
CR2	AnalogBus	CompBus	0	C4				
CR3	0	0	1	0	0	1	电源	

垂直 A 输入拓扑结构

Table 14. 模块 FLIN

位	7	6	5	4	3	2	1	0
CR0	CA	0	极性	C1				
CR1	输入			C2				
CR2	0	0	0	C4				
CR3	0	0	1	0	反馈		电源	

[+] Feedback

Table 15. 模块 FLOUT

位	7	6	5	4	3	2	1	0
CR0	CB	0	0	C3				
CR1	FBIN			0	0	0	0	1
CR2	AnalogBus	CompBus	0	0	0	0	0	0
CR3	0	0	1	0	0	1	电源	

垂直 B 输入拓扑结构

Table 16. 模块 FLIN

位	7	6	5	4	3	2	1	0
CR0	CA	0	1	C2				
CR1	反馈			C1				
CR2	0	0	0	C4				
CR3	0	0	1	0	输入		电源	

Table 17. 模块 FLOUT

位	7	6	5	4	3	2	1	0
CR0	CB	0	0	C3				
CR1	FBIN			0	0	0	0	1
CR2	AnalogBus	CompBus	0	0	0	0	0	0
CR3	0	0	1	0	0	1	电源	

[+] Feedback

可变位字段定义

以下定义适用于上述所有寄存器定义：

CA 和 **CB** 将 FLIN 和 FLOUT 反馈电容分别设置为 16 或 32 个单位（请参见 LPF2 框图）。 CA 和 CB 可以直接在器件编辑器中配置，也可以使用 “ 滤波器设计向导 ” 间接配置。

C1、**C2**、**C3** 和 **C4** 将 LPF2 框图中所示的电容设置为介于 1 和 32 之间的整数值。 与 CA 和 CB 电容类似，它们可以直接在器件编辑器中配置，也可以使用 “ 滤波器设计向导 ” 间接配置。

输入控制用于选择可由 LPF2 用户模块调节的输入信号的复用器。用户模块的 “ 输入 ” 参数决定了此位字段的值。 可以使用器件编辑器手动配置 “ 输入 ” 参数的值。 在某些情况下，此位字段可采用的值将受到限制，这样可保证 FLIN 与 FLOUT 模块之间的 C4 连接能够正常工作。

AnalogBus 用于将滤波器的输出连接至模拟总线。 用户模块的参数 “AnalogBus” 决定了此位字段的值。 可以使用器件编辑器手动配置 AnalogBus 参数的值。

CompBus 用于将滤波器的输出连接至比较器总线。 用户模块的 “CompBus” 参数决定了此位字段的值。 可以使用器件编辑器手动配置 CompBus 参数的值。

反馈是 C2 反馈连接，由 LPF2 用户模块在器件编辑器中的放置位置自动确定。 在某些情况下，此位字段也用于在 FLIN 与 FLOUT 模块之间建立 C4 连接。

FBIN 是 FLIN 输出与 FLOUT 输入的连接，由 LPF2 用户模块在器件编辑器中的放置位置自动确定。

极性控制是否对滤波器的输出进行反相。 可以直接使用器件编辑器配置该位。 此选项仅适用于 A 输入拓扑结构滤波器。

电源控制 PSoC 模块的开 / 关状态以及与当前设置之间的偏差。 该参数最初是通过调用用户模块的 API 函数 LPF2_Start 进行设置的，可通过调用函数 LPF2_SetPower 和 LPF2_Stop. 进行修改。

附录： LPF2 数值滤波器设计

本附录详细介绍了通过设计向导和电子表格实现的设计过程。 所需的值可以轻松地计算出来 （但属于粗略计算）。 对于一阶近似，设计公式简化为以下几个公式：

Equation 8

$$G = -\frac{C_1}{C_2}$$

Equation 9

$$\omega_n\omega_0 = f_{clk}\frac{\sqrt{C_2C_3}}{\sqrt{C_AC_B}}$$

Equation 10

$$d = \frac{C_4}{\sqrt{C_AC_B}}\sqrt{\frac{C_3}{C_2}}$$

这些公式更清晰地表明电容之间比率的一阶交互。

[+] Feedback

时钟频率 f_{clk} 通过对公式 8 进行变换而计算得出：

Equation 11

$$f_{clk} = \omega_n \omega_0 \frac{\sqrt{C_A C_B}}{\sqrt{C_2 C_3}} = 2\pi 10^3 \cdot 1 \cdot \frac{\sqrt{32 \cdot 32}}{\sqrt{2}} = 142.17\ kHz$$

2 极设计过程

该设计过程的目的在于尽可能地产生最高的 f_{clk}，从而获得最佳逼真度并最大限度地减小锯齿：

1. 将 CA 和 CB 设置为 32。
2. 将 C2 设置为最小整数值 1。
3. 将 C4 的初始值设置为最大值 31。
4. 计算出 C3 的最小值，以便满足 $C_3 \geq d^2 C_2 \frac{\sqrt{C_A C_B}}{C_4}$.
5. 根据以下公式修改 C4 $C_4 = d\sqrt{C_A C_B}\sqrt{C_2 / C_3}$ 并将其舍入为最近的整数。
6. 使用所需的增益计算 “ 增益： 公式 4” 中 C1 的值。
7. 根据公式 10 计算 $_{clk}$。
8. 将模拟列时钟设置为 fclk 的四倍。 从 PSoC Designer 中提供的选项中选取模拟列时钟资源 f_{sysclk}。 有关时钟选择的其他详细信息，请参阅此数据手册中的 “ 采样时钟 ” 部分。
9. 计算分频器并将结果舍入为最近的整数：
$$n = int\left(\frac{f_{sysclk}}{4f_{clk}} + 0.5\right)$$
10. 对所选系统时钟资源进行 4n 分频，以获取实际的时钟频率。
11. 根据公式 6 和公式 5 计算阻尼比 d 和固有频率 ω_0。 将结果与所需值进行比较。 这为滤波器设计提供了良好的开端。d 和 ω0 与所需值之间的误差为 -2 到 -10%。
12. 使用公式 ω5 和公式 6 调整 C2、C3 和 C4 的值， 以获得所需的 d 和 0 值。 通常，为了满足 d 要求，首先应减少 C4 的值。
13. 用更新后的电容值重新计算系统时钟频率，以满足 ω_0 要求。

此过程为 LPF2 中的电容生成的值是近似值。 在针对 2 极滤波器的 LPF2 Design 和针对 4 极滤波器的 LPF4 Design 开发电子表格中，将严格根据公式 4、5 和 6 执行更为精确的计算。 在 PSoC Designer 文档目录中列出了这些电子表格。 LPF2 设计通过滤波器向导实现，在器件编辑器中右键单击用户模块即可访问该向导。 在优化滤波器后，可在 PSoC Designer 用户模块参数窗口中输入电容值和时钟频率分频系数。

2 极示例

在此设计示例中，f-3dB 设置为 1.0 kHz。 对于巴特沃斯滤波器，标准化的拐角频率 ω_0 为 1.0，阻尼比 d 为 1.414。 该滤波器设计用于最大过采样率。

该设计过程采用以下数字：

1. CA、CB = 32。
2. C2 = 1。
3. C4 = 31。
4. 计算出非常接近于 2 的 C3。
5. C4 不更改。

[+] Feedback

6. C1 = 1。
7. $$f_{clk} = \omega_n \omega_0 \frac{\sqrt{C_A C_B}}{\sqrt{C_2 C_3}} = 2\pi 10^3 \cdot 1 \cdot \frac{\sqrt{32 \cdot 32}}{\sqrt{2}} = 142.17\ kHz$$
8. 模拟列时钟 = 568.69 kHz。
9. 对于 24V1、24V2 时钟或定时器，或对于计数器 PSoC 模块，n = 42。
10. fclk = 142.857 kHz。

$$d = \frac{C_4\left(\frac{C_3}{C_2}\right)^{\frac{1}{2}}}{\left(C_A C_B + \frac{C_2 C_3}{4} + \frac{C_4 C_3}{2}\right)^{\frac{1}{2}}} = \frac{31\sqrt{2}}{\left(32 \cdot 32 - \frac{2}{4} - \frac{31 \cdot 2}{2}\right)^{\frac{1}{2}}} = 1.392$$

$$\omega_n \omega_0 = \frac{f_s (C_2 C_3)^{\frac{1}{2}}}{\left(C_A C_B - \frac{C_2 C_3}{4} - \frac{C_4 C_3}{2}\right)^{\frac{1}{2}}} = \frac{142.857\ kHz\ \sqrt{2}}{\left(32 \cdot 32 - \frac{2}{4} - \frac{31 \cdot 2}{2}\right)^{\frac{1}{2}}}$$

f-3dB = 1.020 kHz， ω_0 = 1.00 或者

f-3dB = 1.000 kHz， ω_0 = 1.02

11. 对于许多应用程序来说，阻尼比有 1.6% 的误差，拐角频率有 2% 的误差是符合要求的。
12. 这一性能可以提高。通过在定时器上使用 48 MHz 的输入，或将计数器用户模块设置为 85 分频，而不是将系统时钟分频器设置为对 24 MHz 进行 42 分频，可以将 f-3dB 调整为更接近于所需的值。 这样能够产生等于 1.008 kHz 的 f-3dB。

Copyright © 2002-2011 Cypress Semiconductor Corporation. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC Designer™ and Programmable System-on-Chip™ are trademarks and PSoC® is a registered trademark of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.