Four-Pole Low-pass Filter Datasheet LPF4V 2.10

Copyright © 2009-2014 Cypress Semiconductor Corporation. All Rights Reserved.

Resources	PSoC® Blocks			API Memory (Bytes)		Pins (per External I/O)
	Digital	Analog CT	Analog SC	Flash ± 3%	RAM	
CY8C29/28/24xxx, CY8C27x43						
	0	0	4	0	229[1]	1[2]

1. This value is for A-input configuration. For B-input, the configuration value is 217 bytes.

2. It depends on the user module placement and “Input” user property selection.

For one or more fully configured, functional example projects that use this user module go to www.cypress.com/psocexampleprojects

Features and Overview

- User-programmable corner frequency and damping ratio
- Corner frequency 50 Hz to 150 kHz
- Automated design for Bessel, Butterworth, and Chebychev filters
- User-selected over-sample ratio (OSR, ratio of sample frequency to corner frequency)
- Built-in polarity control
- Built-in modulator for use in full-wave detection and frequency translation

The LPF4 User Module uses four switched-capacitor blocks to implement a general-purpose fourth order low-pass filter. Corner frequency and damping ratio are functions of the ratios of programmable on-chip capacitors and clock frequency; no external components are required. Select your filter characteristics and clock frequency; capacitor and clock divider values are automatically calculated in the design tool (wizard). Multiple low-pass filters can be cascaded or combined with band-pass filters to achieve more complex transfer functions.

The filter has programmable gain. The filter is biased at AGND, selected in the Global Resources, and the gain is referred to this level.

DC and AC Electrical Characteristics

The following values are indicative of expected performance and based on characterization data of a standard Butterworth filter: Fcorner = 1.00 kHz, Fsample = 50.0 kHz, Gain =0.0 dB, with capacitor values automatically calculated by PSoC Designer.

Table 1. DC Electrical Performance Characteristics VDD = 5.0 V

Unless otherwise specified, typical values apply at T_J = +25 °C, VDD = 5.00 V, Power = High, Opamp Bias = Low. Limit values apply at T_J = +25 °C, VDD = 5.00 ± 0.50 V, Power = High, Opamp Bias = Low.

Symbol	Description	Conditions	Min	Typ	Max	Units
Vos	Input offset voltage	Default filter	–	20	310	mV
TCVos	Temp coefficient, input offset voltage	Default filter	–	–	50	µV/°C
G-Err1	Error, Gain = 1	Default filter, C1L = 2	–	1.0	1.55	%
Idd_LL	Operating current	Pwr = L, Bias = L	–	0.58	–	mA
Idd_LH		Pwr = L, Bias = H	–	1.12	–	
Idd_ML		Pwr = M, Bias = L	–	2.36	–	
Idd_MH		Pwr = M, Bias = H	–	4.32	–	
Idd_HL		Pwr = H, Bias = L	–	8.48	–	
Idd_HH		Pwr = H, Bias = H	–	17.2	–	

Table 2. DC Electrical Performance Characteristics VDD = 3.3 V

Unless otherwise specified, typical values apply at T_J = +25 °C, VDD = 3.30 V, Power = High, Opamp Bias = Low. Limit values apply at T_J = +25 °C, VDD = 3.30 ±0.33 V, Power = High, Opamp Bias = Low.

Symbol	Description	Conditions	Min	Typ	Max	Units
Vos	Input offset voltage	Default filter	–	20	310	mV
TCVos	Temp coefficient, input offset voltage	Default filter	–	–	50	µV/°C
G-Err1	Error, Gain = 1	Default filter, C1L = 2	–	–	1	%
Idd_LL	Operating current	Pwr = L, Bias = L	–	0.56	–	mA
Idd_LH		Pwr = L, Bias = H	–	1.08	–	
Idd_ML		Pwr = M, Bias = L	–	2.38	–	
Idd_MH		Pwr = M, Bias = H	–	4.20	–	
Idd_HL		Pwr = H, Bias = L	–	8.44	–	
Idd_HH		Pwr = H, Bias = H	–	–	–	

Table 3. AC Electrical Performance Characteristics, VDD = 5.0 V

Unless otherwise specified, typical values apply at TJ = +25 °C, VDD = 5.00 V, Power = High, Opamp Bias = Low. Limit values apply at TJ = +25 °C, VDD = 5.00 ±0.50 V, Power = High, Opamp Bias = Low.

Symbol	Description	Conditions	Min	Typ	Max	Units
f_{ERR}	–3 dB cut-off freq error	Default filter	–	1.0	2.0	%[1]
RespErr	Peak damping	Default filter	–	–	0.5	dB
f_{CLKMAX}	Column clock	Pwr = L, Bias = L	–	–	2.0	MHz
		Pwr = H, Bias = H	–	–	8.0	
Vn_spect	Inband noise	Default filter Pwr = L, Bias = L	–	30	60	µV/rtHz
Vn_int	Integrated noise	Default filter Pwr = L, Bias = L	–	2.5	3.5	mVRMS

Note 1. Scaled to 24.0-MHz clock; verify sysclk ±0.24%.

Table 4. LPF4 AC Electrical Characteristics, VDD = 3.30 V

Unless otherwise specified, typical values apply at T_J = +25°C, VDD = 3.30 V, Power = High, Opamp Bias = Low. Limit values apply at T_J = +25°C, VDD = 3.30 ±0.33 V, Power = High, Opamp Bias = Low.

Symbol	Description	Conditions	Min	Typ	Max	Units
f_{ERR}	–3 dB cut-off freq error	Default filter	–	1.0	2.0	%[1]
RespErr	Peak damping	Default filter	–	–	0.5	dB
f_{CLKMAX}	Column clock	Pwr = L, Bias = L	–	–	2.0	MHz
		Pwr = H, Bias = H	–	–	8.0	
Vn_spect	Inband noise	Default filter Pwr = L, Bias = L	–	30	60	µV/rtHz
Vn_int	Integrated noise	Default filter Pwr = L, Bias = L	–	2.5	3.5	mVRMS

Typical Operating Characteristics

Figure 1. 1.0 dB Chebychev Pass-band, Response OSR = 20

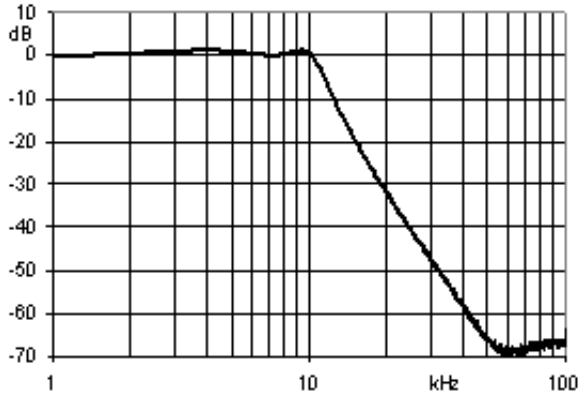


Figure 2. 1.0 dB Chebychev In-Band, Response OSR = 20

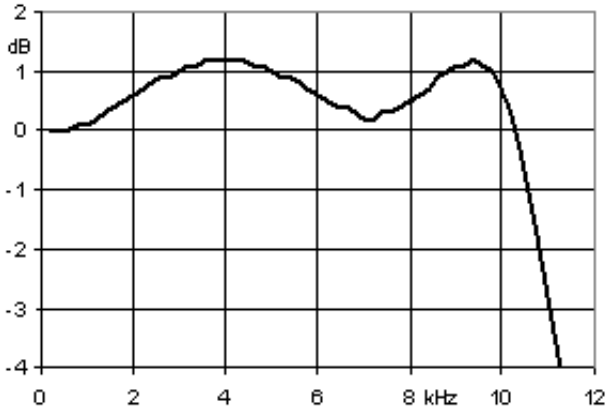


Figure 3. Butterworth Response, OSR = 20

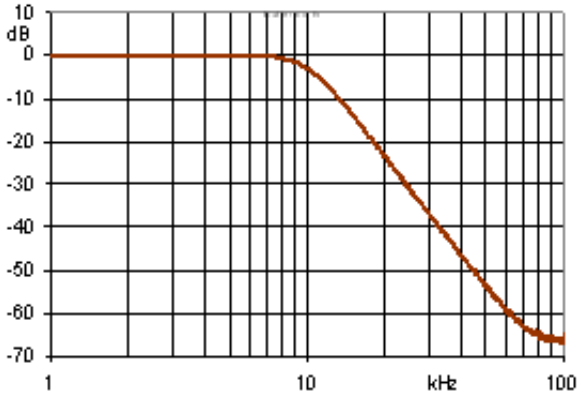


Figure 4. 1.0 dB Chebychev Response, OSR = 20

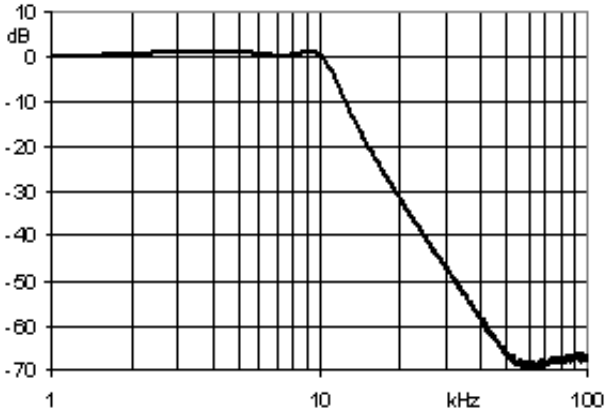
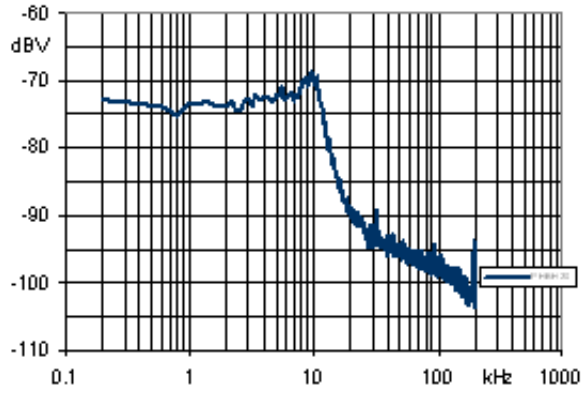


Figure 5. Typical Noise Spectrum,10 kHz 1.0 dB Chebychev, 100 Hz Band, OSR = 20



f_{CORNER} kHz	f_{SAMPLE} kHz	f_{COLUMN} kHz	C1	C2	C3	C4
2.0	200	800	1	1	4	23
2.0	100	400	2	2	7	24
2.0	50	200	4	4	11	25
2.0	25	100	8	8	14	28
2.0	18	72	10	10	16	27
2.0	15	60	12	12	16	28
2.0	12	48	14	14	15	28

Filter Characteristic

In the frequency domain, a four-pole low-pass filter has the transfer function:

Equation 1

$$\frac{V_{OUT}}{V_{IN}} = \frac{Gain(\omega_n\omega_{0L})^2}{s^2 + d_L s\omega_n\omega_{0L} + (\omega_n\omega_{0L})^2} \frac{(\omega_n\omega_{0H})^2}{s^2 + d_H s\omega_n\omega_{0H} + (\omega_n\omega_{0H})^2}$$

where: Gain is the DC gain of the filter

dL and dH are the damping factors for each section

ω0L and ω0H are the natural frequencies for each section

ωn is the normalized corner frequency

s is the Laplace operator = 2* *j*frequency

In-band performance is determined by the damping ratio and natural frequency. Bessel and Butterworth filters are defined as having the –3.0-dB point at the natural frequency. A Chebychev filter has peaking or ripple in the pass-band and gain equal to the DC value at the nominal corner frequency. Values for d and 0 are readily available in any filter design reference. Values for standard forms Bessel, Butterworth, 0.1-dB Chebychev and 1.0-dB Chebychev are built into the design tool.

The filters have a rolloff of 24 dB per octave at frequencies up to half of the sample rate. The recommended OSR is between 6 and 100.

The pass-band performance of standard filter types for a nominal corner frequency of 10 kHz is shown in Figure 6. The 0.1 dB ripple Chebychev has a very flat pass-band, but less attenuation in the near out-of-band than other filters, as seen in Figure 7.

Figure 6. Low-pass Filter In-band Response

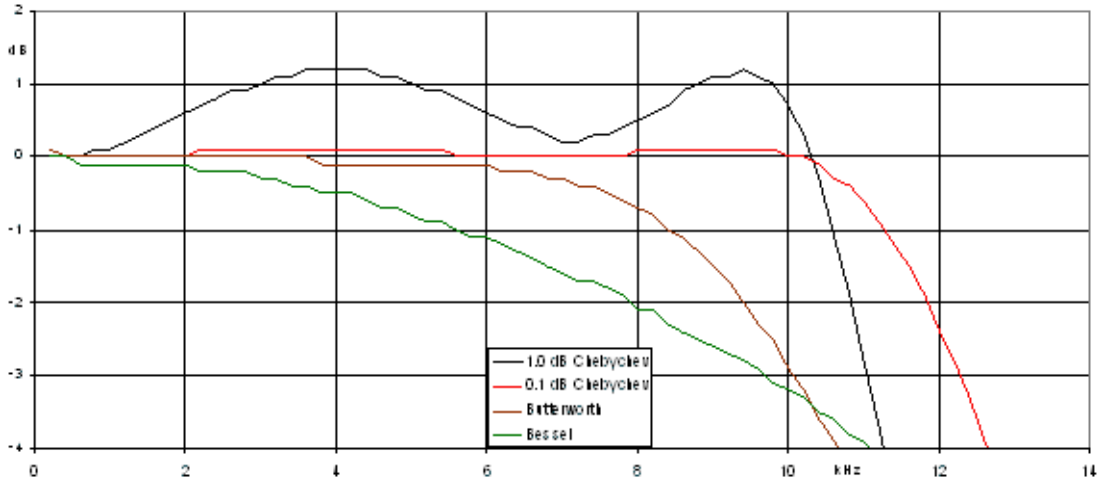
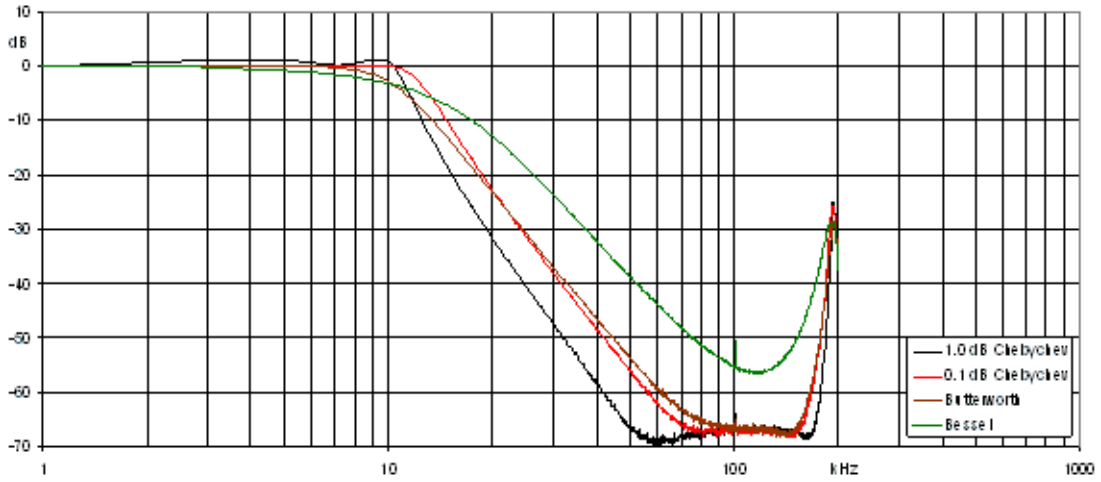


Figure 7. Low-pass Filter Response

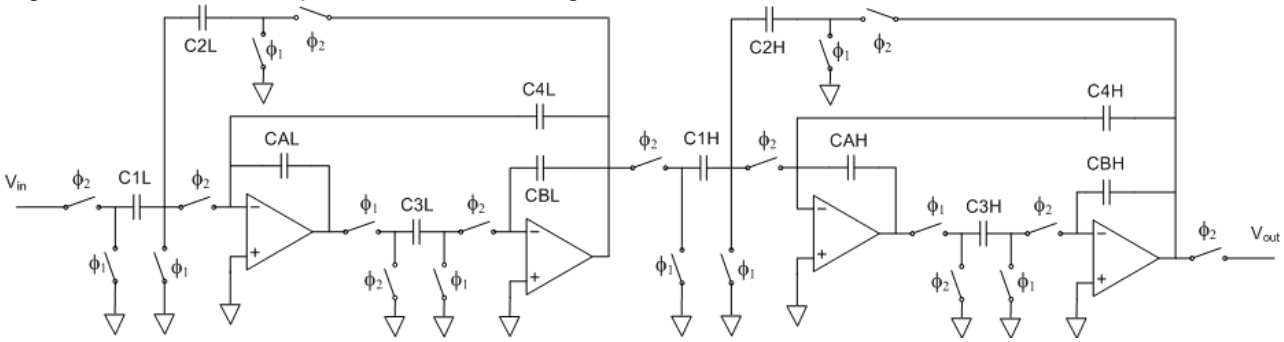


At higher frequencies, the LPF4 switched capacitor filter does not follow the ideal form of the low-pass response. The switching nature of the circuit results in a frequency-dependent gain in the numerator of the equation. This reduces the attenuation at higher frequencies, as shown by the example in Figure 7.

Functional Description

An outline schematic of LPF4 is shown in Figure 1. The filter uses four switched-capacitor blocks and one or two column clocks (at the same frequency) depending on placement. Polarity and modulation inputs are user-selectable.

Figure 8. Switched-Capacitor Filter Block Diagram



A switched-capacitor filter is a time-sampled circuit. The transfer function is derived in the time (z) domain and translated to the frequency (s = jw) domain using the bi-linear transform. The transfer function of the LPF4 is:

Equation 2

$$\frac{Vout}{Vin} = \frac{-\frac{C_{1L}}{C_{2L}}\frac{\left(1-\left(\frac{s}{2f_S}\right)^2\right)f_S^{\,2}}{\left(\frac{C_{4L}C_{BL}}{C_{2L}C_{3L}}-\frac{1}{4}-\frac{1}{2}\frac{C_{4L}}{C_{2L}}\right)}}{s^2+\frac{C_{4L}}{C_{2L}}\frac{sf_S}{\left(\frac{C_{4L}C_{BL}}{C_{2L}C_{3L}}-\frac{1}{4}-\frac{1}{2}\frac{C_{4L}}{C_{2L}}\right)}+\frac{f_S^{\,2}}{\left(\frac{C_{4L}C_{BL}}{C_{2L}C_{3L}}-\frac{1}{4}-\frac{1}{2}\frac{C_{4L}}{C_{2L}}\right)}} \cdot \frac{-\frac{C_{1H}}{C_{2H}}\frac{\left(1-\left(\frac{s}{2f_S}\right)^2\right)f_S^{\,2}}{\left(\frac{C_{4H}C_{BH}}{C_{2H}C_{3H}}-\frac{1}{4}-\frac{1}{2}\frac{C_{4H}}{C_{2H}}\right)}}{s^2+\frac{C_{4H}}{C_{2H}}\frac{sf_S}{\left(\frac{C_{4H}C_{BH}}{C_{2H}C_{3H}}-\frac{1}{4}-\frac{1}{2}\frac{C_{4H}}{C_{2H}}\right)}+\frac{f_S^{\,2}}{\left(\frac{C_{4H}C_{BH}}{C_{2H}C_{3H}}-\frac{1}{4}-\frac{1}{2}\frac{C_{4H}}{C_{2H}}\right)}}$$

where: f_S is the sample rate (= column clock/4)

C1 through CB values are calculated and entered by the user or calculated in the wizard and automatically entered

Mapping equation 3 on to equation 2 yields a set of design equations for each filter section:

Equation 3

$$Gain = \frac{C_1}{C_2}$$

Equation 4

$$(\omega_n \omega_0)^2 = \frac{f_S^{\,2}}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}$$

Equation 5

$$d = \frac{\frac{C_4}{C_2}}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)^{\frac{1}{2}}}$$

Deriving the capacitor and clock values is a non-trivial task, which also includes compensation for frequency distortion due to the bi-linear transform used to translate the transfer function from the original z-domain to the s-domain. An outline of the direct method is included in Appendix 2; the preferred method is to use the automated design procedure in the wizard.

Operational Limits

The maximum corner frequency of the LPF4 is 150 kHz at high power and high bias, determined by opamp slewing and settling. The corner frequency has been tested down to 50 Hz at OSR = 100. The ultimate low-corner frequency may be limited by noise at an elevated temperature and aliasing of external signals. It is advisable to characterize these parameters in your system.

The standard form of a low-pass filter has a response, which goes to zero at infinite frequency. The switched-capacitor filter response in comparison is asymptotic to a finite value. This is a mathematical result (not a circuit "problem") of the (s/2fS)^2 in the numerator of the transfer function. Solving equation 3 for a unity gain filter when frequency (s = 2 f) is very large, the attenuation for a two-pole low-pass filter is:

Equation 6

$$\frac{Vout}{Vin} = \left(\frac{1}{2f_s}\right)^2 \omega_0^2 = \left(\frac{2\pi f_0}{2f_s}\right)^2 = \left(\frac{\pi f_0}{f_s}\right)^2$$

For OSR = 50 (2-kHz filter, 100-kHz sample rate), this is –48 dB at the sample frequency, about 20 dB less attenuation that a non-switched-capacitor filter measured at the same frequency. We expect double this attenuation from a four-pole filter and, in fact, this can be seen in Figure 9 as the response bumps into the limit determined by the filter's noise level. Near in-band performance of the filter is affected by the higher level asymptote at low OSR, resulting in less roll-off. The projected response of the wizard has been tested and demonstrated to agree closely with measured results for a wide range of frequencies and filter types.

Power Setting

The operating current for the LPF4 is set in the user's code using the APIs provided. The operational bandwidth of the switched-capacitor filter is determined by the opamp slew rate and settling time. Slew rate and settling time are a function of opamp gain-bandwidth (GBW), which is a function of the power setting. High power settings result in increased bandwidth. There are a total of six active power settings at VDD = 5.0 V.

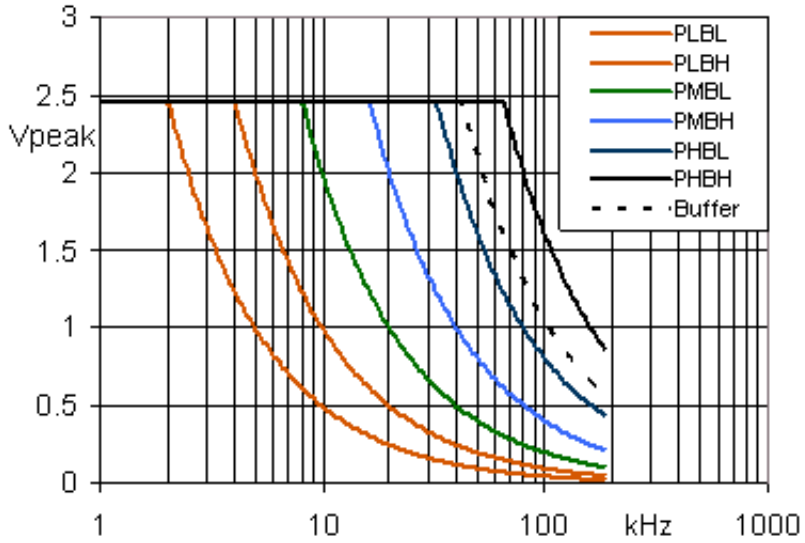
	Power	Bias	Gain*Bandwidth
PLBL	Low	Low	0.75 MHz
PLBH	Low	High	1.5 MHz
PMBL	Medium	Low	2.25 MHz
PMBH	Medium	High	3.1 MHz
PHBL	High	Low	4,5 MHz
PHBH	High	High	9.0 MHz

In this table, P = Power, B = Opamp Bias, L = Low, H = High, and M = Medium.

PHBH (power = high, bias = high) is not available at VDD = 3.3 V. For reliable operation, the GBW should be at least twice the filter corner frequency multiplied by the gain. Excess GBW increases noise and does not improve filter performance.

The ability of the filter to accurately deliver a signal is determined by the slew rate, which is also a function of power and bias setting.

Figure 9. Maximum Signal vs Frequency



The slew rate of the analog column output buffer is 0.65 V/usec. For operation at high power, this results in a lower maximum signal limit than the filter itself.

Placement

The Device Editor maps the logical FLIN and FLOUT blocks onto a pair of adjacent switched capacitor PSoC blocks in the device's analog array. LPF4 UM uses 2 pairs of the FLIN and FLOUT blocks named FLINL, FLOUTL for first pair and FLINH, FLOUTH for second. So LPF4 takes 4 SC blocks per one instance. There are several ways to construct the four-pole low-pass filter circuit out of the analog PSoC blocks. Each construction uses different capacitors and connections within the FLIN and FLOUT blocks. Each results in a different circuit topology with different mapping and I/O consequences. The most noticeable difference is whether the two PSoC blocks lie in a row or column of the analog array. The topologies also determine which connections can be made to other blocks in the array. Regardless of the selected placement, however, the filter inputs always connect to the FLINL block and outputs are driven by the FLOUTH block.

Each time an instance of the LPF4 User Module is created, PSoC Designer presents a dialog with illustrations and text to assist in selecting a circuit topology. The choice may be altered at any time by right-clicking on the user module icon in the selection bar or, if already placed, right-clicking on one of its PSoC blocks and choosing "Select User Module Options..." from the pop-up menu. Changing the topology after placement requires that the user module be placed in the analog array again.

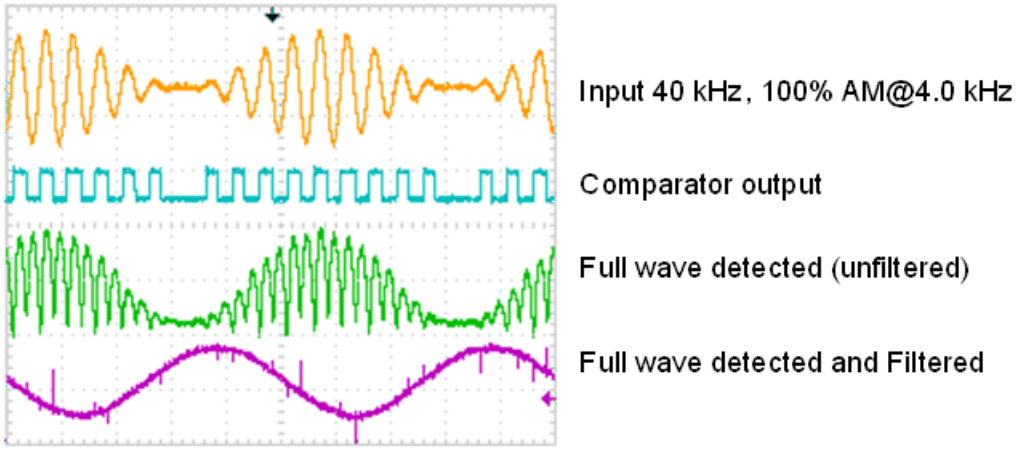
Polarity and Modulator Functions

The input switched-capacitor block has a polarity control parameter. The polarity can also be controlled by a clock external to the block to form a modulator, multiplying the input by +1 or –1 at the clock rate. This generates signals at the sum of the carrier and input frequencies and at the difference of the carrier and input frequencies. Modulation carriers should have 50% duty cycle to minimize even carrier harmonic aliases. The modulation process introduces carrier/signal pairs at odd harmonics of the carrier at a level proportional to 1/n where n is the number of the carrier harmonic. The filter is usually designed to attenuate these out-of-band harmonics.

A common usage of the filter is to generate a sine wave from a digital square wave. This can be done by feeding the filter through a direct Port 2[0..3] input or through a PGA. This results in a sine wave that is dependent on supply voltage. To make the sine wave a fixed level independent of supply, the digital signal can drive the modulator and the input to the filter can be routed to either RefHi or RefLo.

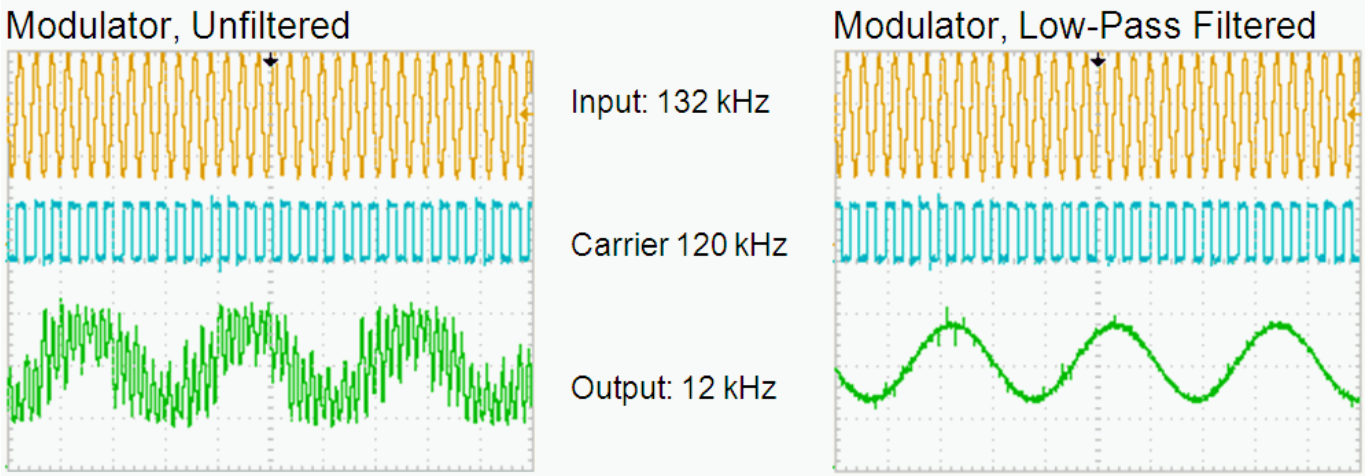
Another common usage of the modulator is a detector. The input fed to the modulator/filter is also fed to a comparator; the digital comparator output drives the modulator input of the filter. The result is full-wave detection. The example shows a 40-kHz input signal, 100% AM modulated at 4.0 kHz.

Figure 10. Full-Wave Detector Waveforms



The modulator can be used to mix frequencies up or down. The example in Figure 12 shows a 132-kHz carrier mixed with a 120-kHz local oscillator driven to the modulator input, resulting in a clean waveform at the difference frequency of 12.0 kHz.

Figure 11. Modulator, Switched-Capacitor Filter Waveforms



Noise

Noise sources in the switched capacitor filter include the opamp, AGND buffer, and capacitor array. The sampling nature of the switched-capacitor filter results in copies of the noise being added at each multiple of the sampling frequency up to the unity gain-bandwidth of the filter's opamps. The best method for reducing aliased noise is to reduce the bandwidth of the opamp; this limits the number of aliased copies of the noise. The example of Figure 10 shows the noise spectrum level for a 1.75 kHz low-pass filter operating at each of the six power levels: Power = Low, Bias = Low (PLBL) through Power = High, Bias = High (PHBH). The noise is integrated across the band to get a total RMS noise level, as shown in Figure 11. Higher power levels have higher integrated noise.

Anti-aliasing

Because of the switching nature of the filter, signals around harmonics of the sample frequency can alias into the pass-band. At the input, a simple single-pole RC low-pass filter with corner frequency above the LPF4's corner frequency and below the sample frequency, can prevent this aliasing and reduce noise. At the output (that is, analog column buffer output for going off-chip), a low-pass filter will attenuate clock noise.

Wizard Parameters and Characteristics

Filter Input Parameters And Characteristics

Figure 12. Standard Filter Approximation

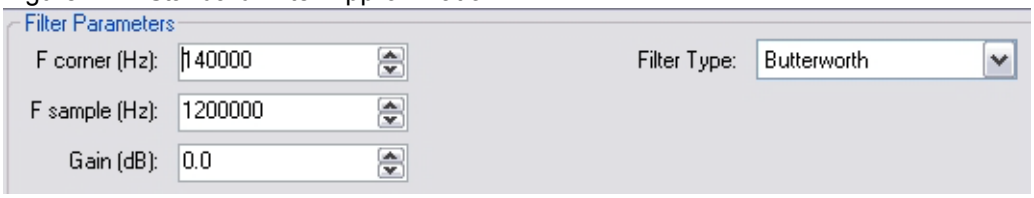


Figure 13. Custom Complex Pole

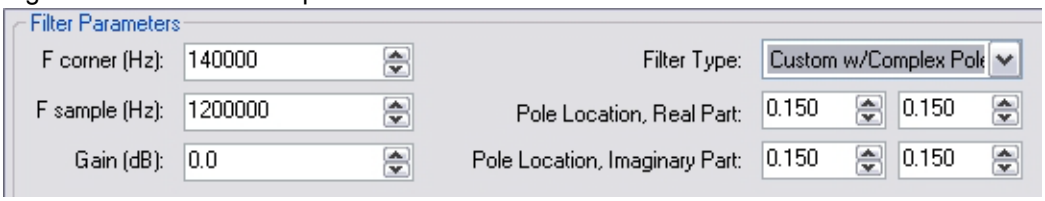


Figure 14. Custom Real Pole



F corner

Enter the desired corner frequency for your filter.

F sample

Enter the desired sample frequency for your filter.

Gain (dB)

Enter the desired gain for your filter in dB.

Filter Type

Select the type of approximation you want to use in your design.

Real Part (Pole Location of)

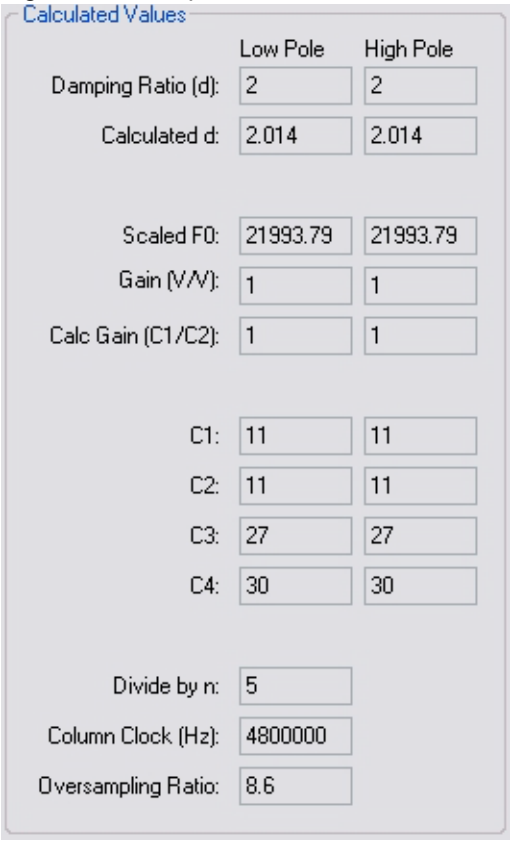
Enter the real part of the filter pole location on a complex plane when customizing filter characteristics.

Imaginary Part (Pole Location of)

Enter the imaginary part of the filter pole location on a complex plane when customizing the filter characteristics.

Filter Output Parameters And Characteristics

Figure 15. Output Parameters



Damping Ratio (d)

Nominal damping ratio of the filter for lower and higher pairs of SC blocks respectively.

Calculated d

Expected damping ratio of the designed filter for lower and higher pairs of SC blocks respectively.

Scaled F0

Expected roll-of frequency of the designed filter for lower and higher pairs of SC blocks respectively.

Gain(V/V)

Requested voltage gain of the filter for lower and higher pairs of SC blocks respectively.

Calc Gain (C1/C2)

Expected voltage gain of the designed filter for lower and higher pairs of SC blocks respectively.

C1

C1 capacitance value of designed filter: The value in the column marked “Low Pole” is transferred to the C1L user module parameter. The value in the column marked “High Pole” is transferred to the C1H user module parameter.

C2

C2 capacitance value of designed filter: The value in the column marked “Low Pole” is transferred to the C2L user module parameter. The value in the column marked “High Pole” is transferred to the C2H user module parameter.

C3

C3 capacitance value of designed filter: The value in the column marked “Low Pole” is transferred to the C3L user module parameter. The value in the column marked “High Pole” is transferred to the C3H user module parameter.

C4

C4 capacitance value of designed filter: The value in the column marked “Low Pole” is transferred to the C4L user module parameter. The value in the column marked “High Pole” is transferred to the C4H user module parameter.

Divide by n

Calculated divider for the filter input clock. Use this divider number when configuring PSoC clock dividers and the column clock where the filter is placed.

Column Clock (Hz)

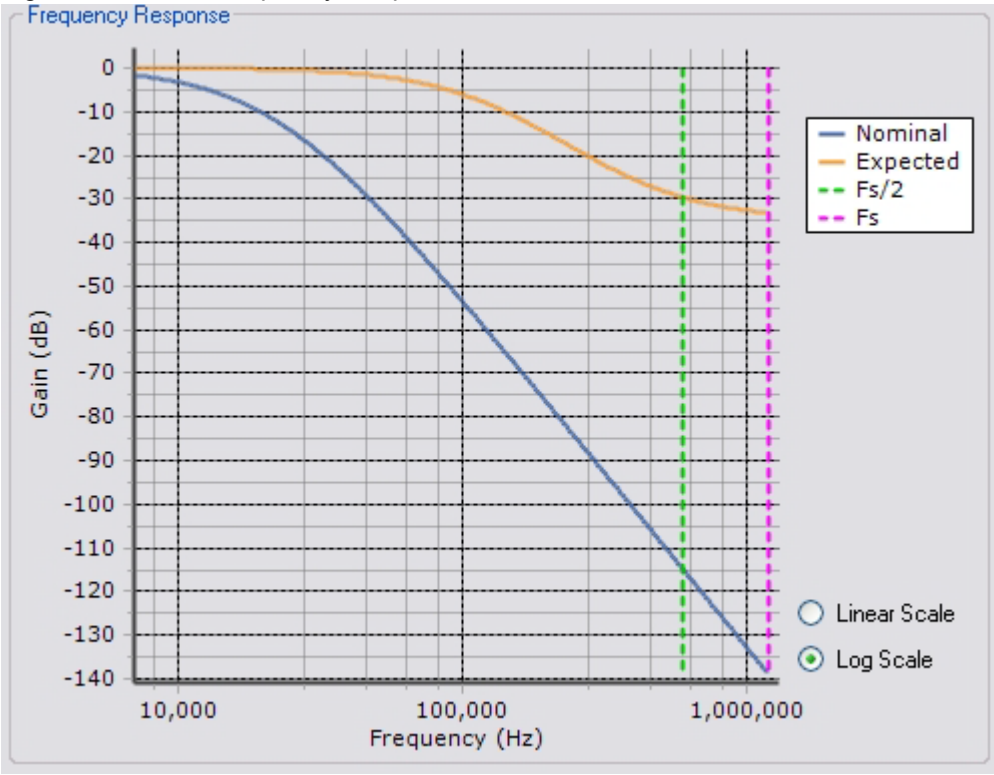
Calculated column clock for the designed filter. Use this value when configuring PSoC clock dividers and the column clock where the filter is placed.

Oversampling Ratio

The oversampling ratio of the designed switched-capacitor filter.

Filter Response Plot

Figure 16. Filter Frequency Response Plot



Frequency response of designed filter

The area where the two frequency response plots are displayed – one for the nominal (desired) filter and one for expected (capable to be implemented on PSoC).

Frequency scale selector

Select the frequency axis scale, which is most representative for you.

Wizard Controls

Figure 17. Configuration Wizard Controls Panel



Help

This button calls this user module datasheet, which has information on how to use the user module wizard. The datasheet's Wizard section describes wizard usage.

OK

This button applies the parameters of the designed filter to the LPF4 User Module if the obtained characteristics meet the requirements of your design.

Print

Press this button if you need to print the Filter Configuration window with all parameters and plot data.

Cancel

This button closes the Filter Configuration window without any modifications of LPF4 User Module.

Parameters and Resources

To make a low-pass filter, place an instance of the LPF4 User Module in the Device Editor's analog array. Use one of the design procedure options to determine the values for the filter's capacitors, then connect the inputs and configure the analog bus connection and clock resources. Each of these parameters are discussed in this section.

Input

Inputs to the filter are driven by the outputs of the adjacent PSoC blocks. You can make input selections in the Device Editor.

AnalogBus

The output of the FLOUTH block can be connected to the analog column output bus. This enables connection to the Analog Output Buffer for the same column and prevents analog output-bus access of other user modules in the same column. All interconnections are configured using the Device Editor.

Capacitor Values C1L, C2L, C3L, C4L, C1H, C2H, C3H and C4H

The ratios of these eight capacitor values determine the frequency and phase response of the filter. The names refer to the capacitors drawn in the LPF4 Schematic Diagram. C1L through C4H take values from 0 to 31 (though values greater than zero are required for meaningful transfer functions). Design of the transfer function may be accomplished using automated or manual procedures. To access the built in design tool, right click on the placed filter and choose "LPF4 Wizard..." from the pop up menu. See the Filter Design section for more information on design.

Modulator Clock (A-Input Topology Filters Only)

This property provides modulator source selection. You can give input modulation signal by this property. Modulator clock source is common for single analog column. But it affects the Input user module's block (FLINL) only. Seven sources are available for user selection, see the following table.

Table 5. Modulator Clock source selection

Parameter	Note
None	Modulator Clock is OFF
GlobalOutEven_0	Get Modulator Clock from GlobalOutOdd_0
GlobalOutEven_1	Get Modulator Clock from GlobalOutOdd_1
Row_0_Broadcast	Get Modulator Clock from Row_0_Broadcast
ComparatorBus_0	Get Modulator Clock from ComparatorBus_0
ComparatorBus_1	Get Modulator Clock from ComparatorBus_1
ComparatorBus_2	Get Modulator Clock from ComparatorBus_2
ComparatorBus_3	Get Modulator Clock from ComparatorBus_3

Sample Frequency (column clock/4)

The required Sample Frequency, equivalent to the column clock divided by 4, for the filter is calculated using the design equations in the Functional Description section. Unlike the other user module parameters, the Sample Frequency does not appear in the list of user module parameters underneath the Device Editors list of Global Resources. In addition, unlike signal inputs that are specific to a particular

user module, the Sample Frequency (column clock/4) serves an entire analog column. The column clocks for all filter PSoC blocks must be the same.

Where horizontal placement of blocks is selected, both column clocks must be driven from the same source to have the same sample frequency. Each column-clock generator divides its input by four to produce φ1 and φ2, the internal clocks in the blocks, so the source must be four times faster than the desired filter sample frequency.

Choices for the clock source include any of the digital PSoC blocks and the system clock dividers. All of the Timer, Counter, and Pulse-Width Modulator (PWM) User Modules are suitable choices when system clock dividers must be consigned to other uses.

The clock source to the column clock is selected using the CLK multiplexer, for each column in the Device Editor. The system clocks are direct inputs to this multiplexer. When PSoC blocks are used for clock generation, they are connected through the ACLK0 and ACLK1 multiplexers to the CLK multiplexer.

Gain

This property is available from the GUI only. It provides gain setting.

Corner Frequency

This property is available from the GUI only. It provides filter corner frequency selection in the range from 20 Hz to 150 kHz.

Filter type

This property is available from the GUI only. It provides selection of classical all-pole filter configurations (Butterworth, Bessel, and Chebyshev). Also user can select custom filter configuration and enter values for real or complex poles.

Application Programming Interface

The Application Programming Interface (API) routines are provided as part of the user module to allow the designer to deal with the module at a higher level. This section specifies the interface to each function together with related constants provided by the “include” files.

Note

In this, as in all user module APIs, the values of the A and X register may be altered by calling an API function. It is the responsibility of the calling function to preserve the values of A and X prior to the call if those values are required after the call. This “registers are volatile” policy was selected for efficiency reasons and has been in force since version 1.0 of PSoC Designer. The C compiler automatically takes care of this requirement. Assembly language programmers must ensure their code observes the policy, too. Though some user module API function may leave A and X unchanged, there is no guarantee they will do so in the future.

For Large Memory Model devices, it is also the caller's responsibility to preserve any value in the CUR_PP, IDX_PP, MVR_PP, and MVW_PP registers. Even though some of these registers may not be modified now, there is no guarantee that will remain the case in future releases.

Entry points are provided to initialize the LPF4 User Module, change power settings, and disable the user module.

LPF4_Start

Description:

Performs all required initialization for this user module and sets the power level for the switched capacitor PSoC blocks.

C Prototype:

```
void  LPF4_Start(BYTE bPowerSetting)
```

Assembly:

```
mov   A, bPowerSetting
lcall  LPF4_Start
```

Parameters:

Symbolic Name	Value
LPF4_LOWPOWER	1
LPF4_MEDPOWER	2
LPF4_HIGHPOWER	3

bPowerSetting: One byte that specifies the power level to both analog PSoC blocks. Following reset and configuration, the PSoC blocks assigned to the instrumentation amplifier are powered down. Symbolic names provided in C and assembly, and their associated values, are given in the following table.

Note For proper performance, filters with corner frequencies above 40 kHz should (1) use LPF4_HIGHPOWER and (2) set the global parameter “Opamp Bias” to High in the Global Parameters window.

Return Value:

None

Side Effects:

The A and X registers may be altered by this function.

LPF4_SetPower

Description:

Sets the power level for the switched capacitor PSoC blocks. May be used to turn the blocks in the user module off and on.

C Prototype:

```
void  LPF4_SetPower(BYTE bPowerSetting)
```

Assembly:

```
mov   A, bPowerSetting
lcall  LPF4_SetPower
```

Parameters:

bPowerSetting: One byte that specifies the power level to both analog PSoC blocks.

Return Value:

None

Side Effects:

The A and X registers may be altered by this function.

LPF4_SetC1L, SetC2L, SetC3L, SetC4L, SetC1H, SetC2H, SetC3H and SetC4H

Description:

Sets the value of specific capacitors in the user module. This allows adjustment of gain by modifying C1, and alteration of filter transfer characteristics by adjusting the other values.

C Prototype:

```
void  LPF4_SetC1L(BYTE bCapValue)
void  LPF4_SetC2L(BYTE bCapValue)
void  LPF4_SetC3L(BYTE bCapValue)
void  LPF4_SetC4L(BYTE bCapValue)
void  LPF4_SetC1H(BYTE bCapValue)
void  LPF4_SetC2H(BYTE bCapValue)
void  LPF4_SetC3H(BYTE bCapValue)
void  LPF4_SetC4H(BYTE bCapValue)
```

Assembly:

```
mov   A, bCapValue
lcall  LPF4_SetC1L                ; or, call  LPF4_SetC2L (or SetC3L or SetC4L or
SetC1H etc.)
```

Parameters:

bCapValue: Integer value from 0 to 31 for C1L, C2L, C3L, C4L, C1H, C2H, C3H and C4H (see the LPF4 Schematic Drawing). Values outside this range will be truncated to 31.

Return Values:

None

Side Effects:

The A and X registers may be altered by this function.

LPF4_Stop

Description:

Powers the user module off.

C Prototype:

```
void  LPF4_Stop(void)
```

Assembly:

```
lcall  LPF4_Stop
```

Parameters:

None

Return Value:

None

Side Effects:

The A and X registers may be altered by this function.

Sample Firmware Source Code

In C, using the low-pass filter is as simple as using the Start API to begin operation and calling the Stop API when done.

```
//
//  This sample shows how to create a Low Pass Filter with corner frequency 1kHz.
//
// OVERVIEW:
//
//  The LPF4 input/output can be routed to any analog pin or adjacent analog block
//  depending on placement.
//  In this example the LPF input is routed to P0[5] and the output is routed to P0[3].
//
//The following changes need to be made to the default settings in the Device Editor:
//
//   1. Choose the LPF4AS MUM configuration of the LPF4 user module.
//   2. Place it onto ASC10, ASD11, ASD20 and ASC21 blocks.
//   3. Rename the User Module's instance name to LPF4.
//   4. Run the LPF2 Wizard from the context menu.
//      - Set the F corner (Hz) parameter to 1000
//      - Set the F sample (Hz) parameter to 100000
//      - Leave the Gain and Filter Type parameters by default
//       ("0.0" and "Butterworth" respectively)
//      - Click the "OK" button
//   5. Set the LPF4's Input parameter to ACB00.
//   6. Set the LPF4's AnalogBus parameter to AnalogOutBus_0.
//   7. Leave the rest of UM parameters by default.
//   8. Set the AnalogColumn_Clock_0 to VC2 (on the interconnect view)
//   9. Set the AnalogOutBuf_0 to Port_0_3 (on the interconnect view).
//   10. Place the PGA UM onto ACB00 block.
//   11. Rename the User Module's instance name to PGA.
//   12. Set the PGA's Gain parameter to 1.000.
//   13. Set the PGA's Input parameter to AnalogColumn_InputMUX_0.
//   14. Set the PGA's Reference parameter to AGND.
//   15. Set the PGA's AnalogBus parameter to Disable.
//   16. Set the AnalogColumn_InputMux_0 to Port_0_5 (on the interconnect view)
//
// CONFIGURATION DETAILS:
//
//   1. The UM's instance names have to be shortened to LPF4 and PGA.
//   2. The Analog Column clock should be 400 kHz to get the 1kHz low Pass Filter with
//      Over Sampling Ration = 100
//
// PROJECT SETTINGS:
//
//   1. Set the VC1=SysClk/N to 15
//   2. Set the VC2=VC1/N to 4
//   3. Set the A_Buff_Power to High
//
//
// USER MODULE PARAMETER SETTINGS:
```

```
//
//  ------------------------------------------------------------------------------
//  UM          Parameter          Value                   Comments
//  ------------------------------------------------------------------------------
//  LPF4        Name               LPF4                    UM's instance name
//              Input              ACB00
//              C1L                1                       Set by Wizard
//              C2L                1                       Set by Wizard
//              C3L                4                       Set by Wizard
//              C4L                29                      Set by Wizard
//              C1H                2                       Set by Wizard
//              C2H                2                       Set by Wizard
//              C3H                2                       Set by Wizard
//              C4H                24                      Set by Wizard
//              Modulator Clock    None                    Default value
//              AnalogBus          AnalogOutBus_0
//
// PGA          Name               PGA                     UM's instance name
//              Gain               1.000
//              Input              AnalogColumn_InputMUX_0
//              Reference          AGND
//              AnalogBus          Disable
//  ------------------------------------------------------------------------------

/* Code begins here */

#include "PSoCAPI.h"    // PSoC API definitions for all User Modules
#include <m8c.h>        // part specific constants and macros

void main(void)
{
    // M8C_EnableGInt ;            // Uncomment this line to enable Global Interrupts
    PGA_Start(PGA_HIGHPOWER);      // Turn on the PGA
    LPF4_Start(LPF4_HIGHPOWER);    // Turn on the LPF
}
```

Equivalent assembly language code is:

```
;
;  This sample shows how to create a Low Pass Filter with corner frequency 1kHz.
;
; OVERVIEW:
;
;  The LPF4 input/output can be routed to any analog pin or adjacent analog block
;  depending on placement.
;  In this example the LPF input is routed to P0[5] and the output is routed to P0[3].
;
;The following changes need to be made to the default settings in the Device Editor:
;
;   1. Choose the LPF4AS MUM configuration of the LPF4 user module.
;   2. Place it onto ASC10, ASD11, ASD20 and ASC21 blocks.
;   3. Rename the User Module's instance name to LPF4.
;   4. Run the LPF2 Wizard from the context menu.
;      - Set the F corner (Hz) parameter to 1000
;      - Set the F sample (Hz) parameter to 100000
```

```
;       - Leave the Gain and Filter Type parameters by default
;         ("0.0" and "Butterworth" respectively)
;       - Click the "OK" button
;    5. Set the LPF4's Input parameter to ACB00.
;    6. Set the LPF4's AnalogBus parameter to AnalogOutBus_0.
;    7. Leave the rest of UM parameters by default.
;    8. Set the AnalogColumn_Clock_0 to VC2 (on the interconnect view)
;    9. Set the AnalogOutBuf_0 to Port_0_3 (on the interconnect view).
;    10. Place the PGA UM onto ACB00 block.
;    11. Rename the User Module's instance name to PGA.
;    12. Set the PGA's Gain parameter to 1.000.
;    13. Set the PGA's Input parameter to AnalogColumn_InputMUX_0.
;    14. Set the PGA's Reference parameter to AGND.
;    15. Set the PGA's AnalogBus parameter to Disable.
;    16. Set the AnalogColumn_InputMux_0 to Port_0_5 (on the interconnect view)
;
; CONFIGURATION DETAILS:
;
;    1. The UM's instance names have to be shortened to LPF4 and PGA.
;    2. The Analog Column clock should be 400 kHz to get the 1kHz low Pass Filter with
;       Over Sampling Ration = 100.
;
; PROJECT SETTINGS:
;    1. Set the VC1=SysClk/N to 15
;    2. Set the VC2=VC1/N to 4
;    3. Set the A_Buff_Power to High
;
;
; USER MODULE PARAMETER SETTINGS:
;
;   ------------------------------------------------------------------------------
;   UM          Parameter          Value                 Comments
;   ------------------------------------------------------------------------------
;   LPF4        Name               LPF4                  UM's instance name
;               Input              ACB00
;               C1L                1                     Set by Wizard
;               C2L                1                     Set by Wizard
;               C3L                4                     Set by Wizard
;               C4L                29                    Set by Wizard
;               C1H                2                     Set by Wizard
;               C2H                2                     Set by Wizard
;               C3H                2                     Set by Wizard
;               C4H                24                    Set by Wizard
;               Modulator Clock    None                  Default value
;               AnalogBus          AnalogOutBus_0
;
; PGA           Name               PGA                   UM's instance name
;               Gain               1.000
;               Input              AnalogColumn_InputMUX_0
;               Reference          AGND
;               AnalogBus          Disable
;   ------------------------------------------------------------------------------

; Code begins here
```

```
include "m8c.inc"       ; part specific constants and macros
include "memory.inc"    ; Constants & macros for SMM/LMM and Compiler
include "PSoCAPI.inc"   ; PSoC API definitions for all User Modules

export _main

_main:

    ; M8C_EnableGInt ; Uncomment this line to enable Global Interrupts
    mov   A, PGA_HIGHPOWER
    lcall PGA_Start
    mov   A, LPF4_HIGHPOWER
    lcall LPF4_Start

    ; Insert your main assembly code here.

.terminate:
    jmp .terminate
```

Note The design equations show that gain is proportional to the value of C1, but the corner frequency and damping do not depend on it. Once the transfer function is chosen, the LPF4_SetC1L API function may be used to implement a programmable-gain control.

Configuration Registers

The topology and placement of the LPF4 User Module determines over half the bits in the configuration registers for the analog switched capacitor PSoC blocks used. Of those, the ones that are independent of placement location are indicated by fixed values in the register tables. Of the variable bitfields, most are determined by selection of input and transfer function design. Definitions of the variable bitfields used in the register definitions follow, at the end of this section.

Horizontal A-Input Topology

Table 6. Block FLINL: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CAL	0	Polarity	C1L				
CR1	Input			C2L				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	Feedback		Power	

Horizontal B-Input Topology

Table 7. Block FLINL: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CAL	0	1	C2L				
CR1	Feedback			C1L				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	Input		Power	

Table 8. Block FLOUTL: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CBL	0	0	C3L				
CR1	FBINL			0	0	0	0	1
CR2	0	0	0	C4L				
CR3	0	0	1	0	0	1	Power	

Table 9. Block FLINH: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CAH	0	1	C2H				
CR1	Feedback			C1H				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	FLOUTL		Power	

Table 10. Block FLOUTH: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CBH	0	0	C3H				
CR1	FBINH			0	0	0	0	1
CR2	AnalogBus	CompBus	0	C4H				
CR3	0	0	1	0	0	1	Power	

Vertical A-Input Topology

Table 11. Block FLINL: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CAL	0	Polarity	C1L				
CR1	Input			C2L				
CR2	0	0	0	C4L				
CR3	0	0	1	0	Feedback		Power	

Table 12. Block FLOUTL: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CBL	0	0	C3L				
CR1	FBINL			0	0	0	0	1
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	0	1	Power	

Table 13. Block FLINH: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CAH	0	Polarity	C1H				
CR1	FLOUTL			C2H				
CR2	0	0	0	C4H				
CR3	0	0	1	0	Feedback		Power	

Table 14. Block FLOUTH: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CBH	0	0	C3H				
CR1	FBINH			0	0	0	0	1
CR2	AnalogBus	CompBus	0	0	0	0	0	0
CR3	0	0	1	0	0	1	Power	

Variable BitField Definitions

The following definitions apply to all preceding register definitions.

CAL, **CAH, CBL** and **CBH** set the FLIN and FLOUT feedback capacitors, respectively, to either 16 or 32 units. (See the LPF4 Schematic Diagram.)

C1L, **C2L**, **C3L**, **C4L, CPPL, C1H**, **C2H**, **C3H and C4H** set the capacitors illustrated in the LPF4 Schematic Diagram to integer values between 1 and 31. They are configured directly in the Device Editor or indirectly through use of the filter Design Wizard.

Input controls the multiplexor that selects the input signal to be conditioned by the LPF4 User Module.The user module “Input” parameter determines the value of this bitfield. The value of the Input parameter is manually configured using the Device Editor.

AnalogBus enables connection of the filter output to the analog bus. The user module “AnalogBus” parameter determines the value of this bitfield. The value of the AnalogBus parameter is manually configured using the Device Editor.

CompBus enables connection of the filter output to the comparator bus. This property is not used in current UM.

Feedback is the C2L and C2H feedback connection, automatically determined by placement of the LPF4 User Module in the Device Editor.

FBINL is the connection from the FLINL output to the FLOUTL input, automatically determined by placement of the LPF4 User Module in the Device Editor.

FLOUTL is the connection from the FLOUTL output to the FBINH input, automatically determined by placement of the LPF4 User Module in the Device Editor.

FBINH is the connection from the FLINH output to the FLOUTH input, automatically determined by placement of the LPF4 User Module in the Device Editor.

Polarity controls whether the output of the filter is inverted or not. This property is not used in current UM

Power controls the On/Off state of the PSoC block and bias current setting. It is set initially by calling the user module API function LPF4_Start and can be modified by calling the functions LPF4_SetPower and LPF4_Stop.

Appendix: Numerical Filter Design for the LPF4

This appendix details the design procedures implemented in the design wizard and spreadsheets. The required values can be calculated easily (but roughly). For a first order approximation, the design equations are simplified to the following.

Equation 7

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

Equation 8

$$d_X = \frac{C_{4_X}\left(\frac{C_{3_X}}{C_{2_X}}\right)^{\frac{1}{2}}}{\left(C_{A_X}C_{B_X} - \frac{C_{2_X}C_{3_X}}{4} - \frac{C_{4_X}C_{3_X}}{2}\right)^{\frac{1}{2}}}$$

Equation 9

$$d = \frac{C_4}{\sqrt{C_A C_B}}\sqrt{\frac{C_3}{C_2}}$$

These equations show the first order interaction of the ratios of the capacitors clearly.

The clock frequency, fclk, is calculated by rearranging Equation 10:

Equation 10

$$f_{clk} = \omega_n \omega_0 \frac{\sqrt{C_A C_B}}{\sqrt{C_2 C_3}}$$

Four-pole Design Procedure

The design procedure is provided for both biquad parts of 4-pole filter with the same algorithm. There is the algorithm for one of them described below.

The objective of the design procedure is to yield the highest possible fclk, for the best fidelity and least aliasing.

1. Set CA and CB to 32.
2. Set C2 to the minimum integer value, 1.
3. Set the initial value of C4 to the maximum value, 31.
4. Find the smallest value for C3 such that $C_3 \geq d^2 C_2 \frac{\sqrt{C_A C_B}}{C_4}$
5. Revise C4 from $C_4 = d\sqrt{C_A C_B}\sqrt{C_2 / C_3}$ and round to the nearest integer.
6. Using the required gain, calculate a value for C1 from Gain: Equation 6.
7. Calculate fclk from Equation 12.
8. Set analog column clocks to four times fclk. Pick an analog column clock resource, fsysclk, from the selections available in PSoC Designer. See the Sample Clock section in this datasheet, for additional details on clock selection.
9. Calculate the divider by rounding to the nearest integer as follows. $n = \text{int}\left(\frac{f_{sysclk}}{4 f_{clk}} + 0.5\right)$
10. Divide the selected system clock resource by 4n, to get the actual clock frequency.
11. Calculate the damping ratio, d, and the natural frequency, ω0, from Equation 8 and Equation 7. Compare the results to the desired values. This provides an adequate starting point for the filter design. d and ω0 are both be in error by -2 to -10% from the required values.
12. Adjust the values of C2, C3 and C4 to achieve the required values of d and ω0 using Equation 7 and Equation 8. In general, start by reducing C4 to meet the d requirement.
13. Recalculate system clock frequency to meet ω0 requirements with updated capacitor values.

This procedure yields approximate values for the capacitors in the LPF4. More exact calculations, with close adherence to design equations 4, 5 and 6, are executed in the development spreadsheets LPF4 Design for four-pole filters and LPF4 Design for four-pole filters. These spreadsheets are available in the PSoC Designer documentation directory. LPF4 design is implemented in a filter wizard accessible by right-clicking on the user module in the Device Editor. After filter optimization, the capacitor values and clock frequency divisor coefficients can be entered in the PSoC Designer User Module Parameter window.

Four-Pole Example

For this design example, f-3dB is set to 1.0 kHz. For the Butterworth filter, the normalized corner frequency, ω0, is 1.0 and the damping ratio, d, is 1.414. The filter is designed for the maximum over-sample ratio.

The design procedure is followed by the numbers:

1. CA, CB = 32.
2. C2 = 1.
3. C4 = 31.
4. Calculate C3 very close to 2.
5. No change to C4.
6. C1 = 1.
7. $$f_{clk} = \omega_n \omega_0 \frac{\sqrt{C_A C_B}}{\sqrt{C_2 C_3}} = 2\pi 10^3 \cdot 1 \cdot \frac{\sqrt{32 \cdot 32}}{\sqrt{2}} = 142.17\ \text{kHz}$$
8. Analog Column Clock = 568.69 kHz.
9. n = 42 from 24V1, 24V2 clock or Timer, or Counter PSoC block.
10. fclk = 142.857 kHz.

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

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

f-3dB = 1.020 kHz and ω0 = 1.00 orf-3dB = 1.000 kHz and ω0

11. For many applications, 1.6% error in damping ratio and 2% error in corner frequency are adequate.
12. This performance can be improved. f-3dB can be adjusted closer to the desired value by using the 48 MHz input to a Timer or Counter User Module set to divide by 85, rather than setting the system clock dividers to divide 24 MHz by 42. This yields f-3dB equal to 1.008 kHz.

Version History

Version	Originator	Description
2.0	DHA	Replaced the "GlobalOutOdd_0" and "GlobalOutOdd_1" values of the "Modulator Clock" Property with "GlobalOutEven_1" and "GlobalOutEven_0".
2.10	DHA	1. Added wizard help button. 2. Updated the images in this user module datasheet.
2.10.b	MYKZ	Users can now store printer settings in User Module Wizard.

Note PSoC Designer 5.1 introduces a Version History in all user module datasheets. This section documents high level descriptions of the differences between the current and previous user module versions.

Copyright © 2009-2014 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.