Two-Pole Low-Pass Filter Datasheet LPF2 V 4.10

Resources	PSoC® Blocks			API Memory (Bytes)		Pins (per External I/O)
	Digital	Analog CT	Analog SC	Flash	RAM	
CY8C29/27/24xxx, CY8C23x33, CY8CLED04/08/16, CY8CLED0xD, CY8CLED0xG, CY8CTMG120, CY8CTMA120, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, CY8C28x52						
	0	0	2	120	0	1

Features and Overview

- User-programmable corner frequency and damping ratio
- Corner frequency ranging from 20 Hz to 180 kHz
- Automated design for Bessel, Butterworth, and Chebychev filters
- User-selected oversample 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 LPF2 User Module uses two switched-capacitor blocks to implement a general-purpose second 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. Selects 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 initial characterization data. Unless otherwise specified, all limits guaranteed for TJ =+25 °C, VDD = 5.0 V, Power HIGH, Opamp bias LOW, output referenced to Analog Ground = 2*VBandGap. The default filter for tests is 2.0 kHz Butterworth, sample rate = 200 kHz (column clock = 800 kHz), gain = 1, and modulation OFF.

Table 1. DC Electrical Performance Characteristics, V_{DD} = 5.0 V

Symbol	Description	Conditions	Min	Typ	Max	Units
Vos	Input offset voltage	Default filter	–	–	140	mV
TCVos	Temp coefficient, input offset voltage	Default filter	–	–	50	µV/°C
G-Err1	Error, Gain = 1	Default filter, C1 = 1	–	–	1	%
G_Err2	Error, Gain = 2	Default filter, C1 = 2	–	–	1.5	
G_Err4	Error, Gain = 4	Default filter, C1 = 4	–	–	2.0	
Idd_LL	Operating current	Pwr = L, Bias = L	–	–	–	
Idd_LH		Pwr = L, Bias = H	–	–	–	
Idd_ML		Pwr = M, Bias = L	–	–	–	
Idd_MH		Pwr = M, Bias = H	–	–	–	
Idd_HL		Pwr = H, Bias = L	–	–	–	
Idd_HH		Pwr = H, Bias = H	–	–	–	

Table 2. DC Electrical Performance Characteristics, V_{DD} = 3.3 V

Symbol	Description	Conditions	Min	Typ	Max	Units
Vos	Input offset voltage	Default filter	–	–	140	mV
TCVos	Temp coefficient, input offset voltage	Default filter	–	–	50	µV/°C
G-Err1	Error, Gain = 1	Default filter, C1 = 1	–	–	1	%
G_Err2	Error, Gain = 2	Default filter, C1 = 2	–	–	1.5	
G_Err4	Error, Gain = 4	Default filter, C1 = 4	–	–	2.0	
Idd_LL	Operating current	Pwr = L, Bias = L	–	–	–	
Idd_LH		Pwr = L, Bias = H	–	–	–	
Idd_ML		Pwr = M, Bias = L	–	–	–	
Idd_MH		Pwr = M, Bias = H	–	–	–	
Idd_HL		Pwr = H, Bias = L	–	–	–	
Idd_HH		Pwr = H, Bias = H	–	–	–	

Table 3. AC Electrical Performance Characteristics, V_{DD} = 5.0 V (*Default filter for tests is 2.0 kHz Butterworth, sample rate = 200 kHz (column clock = 800 kHz), gain = 1, modulation OFF*)

Symbol	Description	Conditions	Min	Typ	Max	Units
f_{CORNER}	Corner frequency	Min, OSR = 100 Max, OSR = 6	1	–	160	kHz
f_{ERR100}	–3 dB cut-off freq error	2.0 kHz Butterworth, OSR = 100[1]	–	–	1.0	%[3]
f_{ERR10}	–3 dB Cut-off freq error	2.0 kHz Butterworth, OSR = 10[2]	–	–	5.0	%[3]
RespErr	Peak damping	2.0 kHz Chebychev, OSR = 100[4]	–	–	0.5	dB
		2.0 kHz Chebychev, OSR = 10[5]	–	–	1.5	
THD	Total harmonic distortion	f_{IN} = 400 Hz, V_{IN} = 4 Vpp	–	–63	–	dB
Vn_spect	Inband noise	Pwr = L, Bias = L	–	–	–	nV/rtHz
		Pwr = H, Bias = H	–	–	–	
Vn_int	Integrated noise	Pwr = L, Bias = L	–	–	–	nV/rtHz
		Pwr = H, Bias = L	–	–	–	

Note

1. f_{ERR100} Butterworth design C1=1, C2=1, C3=4, C4=23, CA=32,CB=32
2. f_{ERR10} Butterworth design C1=8, C2=8, C3=17, C4=19, CA=32,CB=16
3. Scaled to 24.0 MHz clock, verify sysclk +/- 0.25%
4. f_{ERR100} Chebychev design C1=1, C2=1, C3=4, C4=16, CA=32, CB=32
5. f_{ERR10} Chebychev design C1=9, C2=9, C3=16, C4=16, CA=32, CB=16

Table 4. AC Electrical Performance Characteristics, V_{DD} = 3.3V (*Default filter for tests is 2.0 kHz Butterworth, sample rate = 200 kHz (column clock = 800 kHz), gain = 1, modulation OFF*)

Symbol	Description	Conditions	Min	Typ	Max	Units
f_{CORNER}	Corner frequency	Min, OSR = 100 Max, OSR = 6	1	–	80	kHz
f_{ERR100}	–3 dB cut-off freq error	2.0 kHz Butterworth, OSR = 100[1]	–	–	1.0	%[3]
f_{ERR10}	–3 dB Cut-off freq error	2.0 kHz Butterworth, OSR = 10[2]	–	–	5.0	%[3]
RespErr	Peak damping	2.0 kHz Chebychev, OSR = 100[4]	–	–	0.5	dB
		2.0 kHz Chebychev, OSR=10[5]	–	–	1.5	
THD	Total harmonic distortion	f_{IN} = 400 Hz, VIN= 4 Vpp	–	–	–	dB
Vn_spect	Inband noise	Pwr = L, Bias = L	–	–	–	nV/rtHz
		Pwr = H, Bias = H	–	–	–	
Vn_int	Integrated noise	Pwr = L, Bias = L	–	–	–	nV/rtHz
		Pwr = H, Bias = L	–	–	–	

Typical Operating Characteristics

Figure 1. Butterworth Pass-band Response, OSR = 10

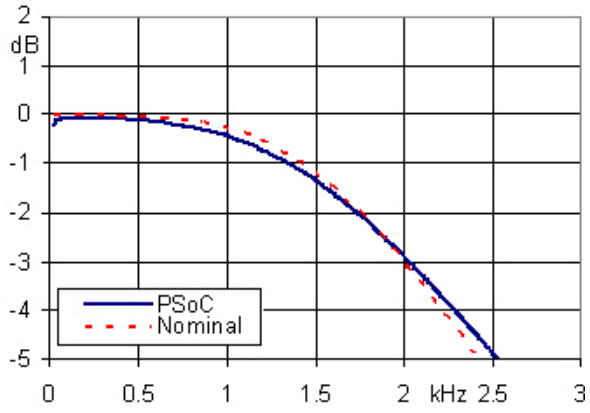


Figure 2. 1.0 dB Chebychev Pass-Band, Response OSR = 10

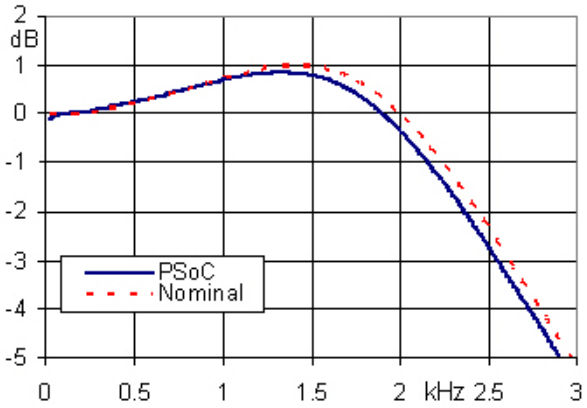


Figure 3. Butterworth Pass-band Response, OSR = 100

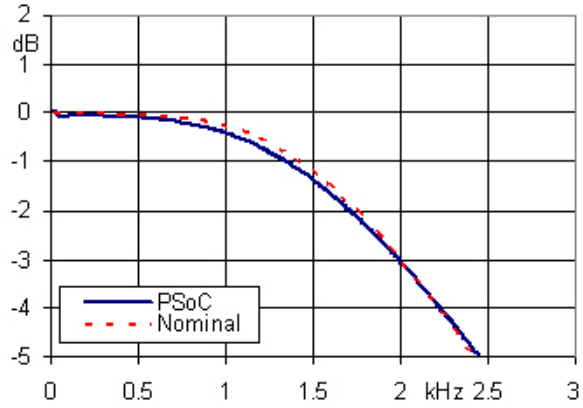


Figure 4. Chebychev Pass-band Response, OSR = 100

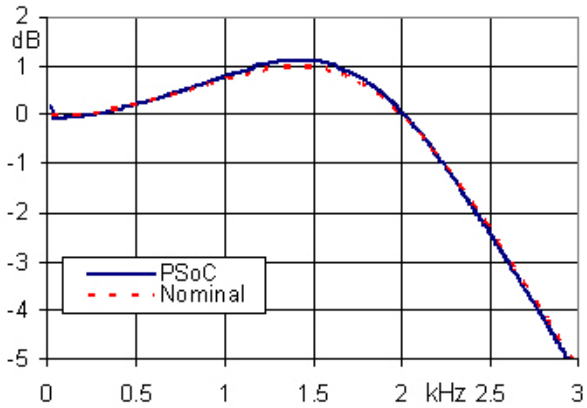


Figure 5. Butterworth Response, OSR = 10

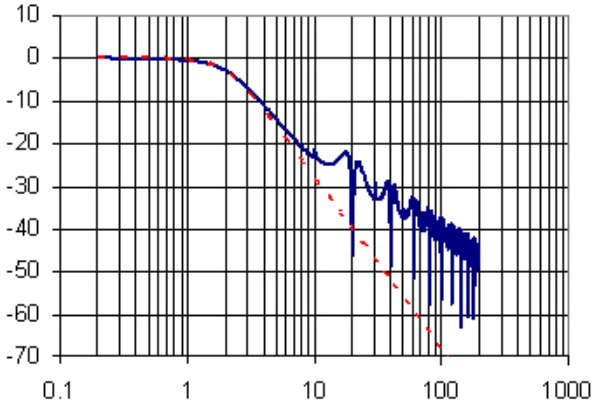


Figure 6. Chebychev Response, OSR = 10

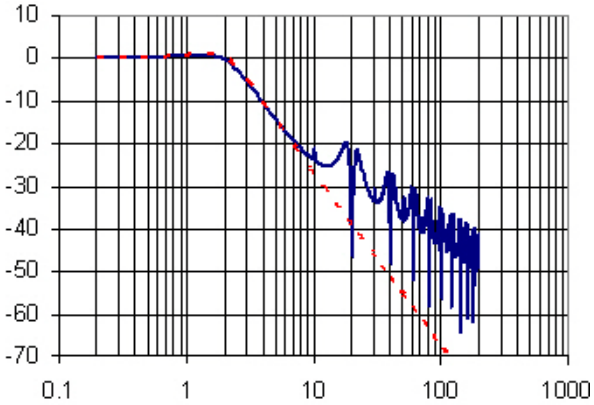


Figure 7. Butterworth Response, OSR = 100

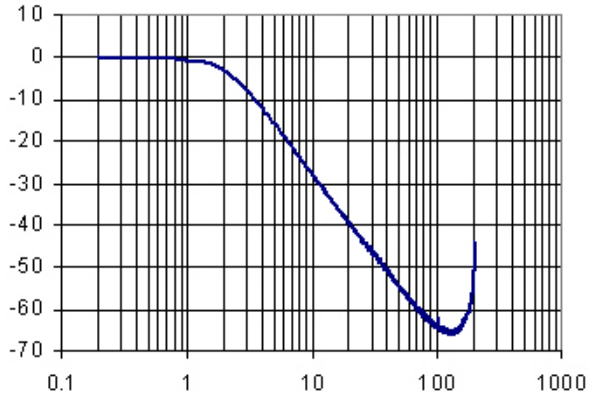


Figure 8. Chebychev Response, OSR = 100

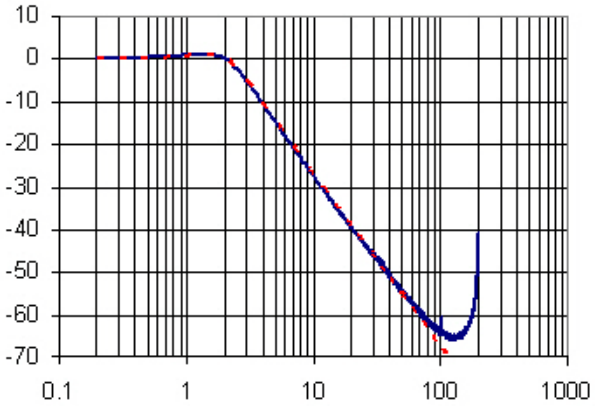


Figure 9. Typical Noise Spectrum, 10 kHz Filter, OSR = 100

TBD

Figure 10. Typical Noise Spectrum, 100 kHz Filter, OSR = 20

TBD

Figure 11. Noise Spectrum at Power Setting

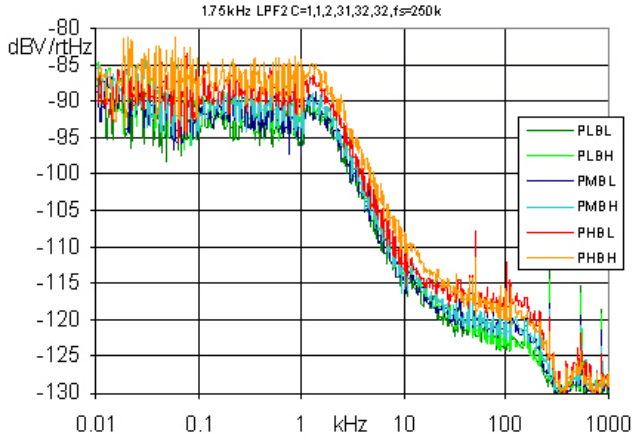


Figure 12. Integrated Noise at Power Setting

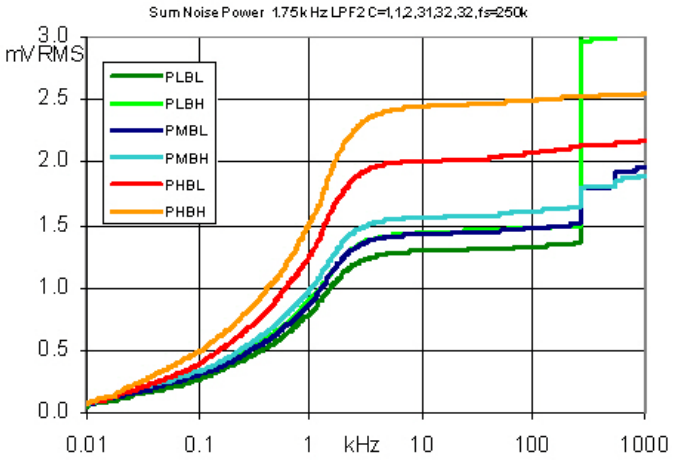


Figure 13. Typical Vos vs Cap Value

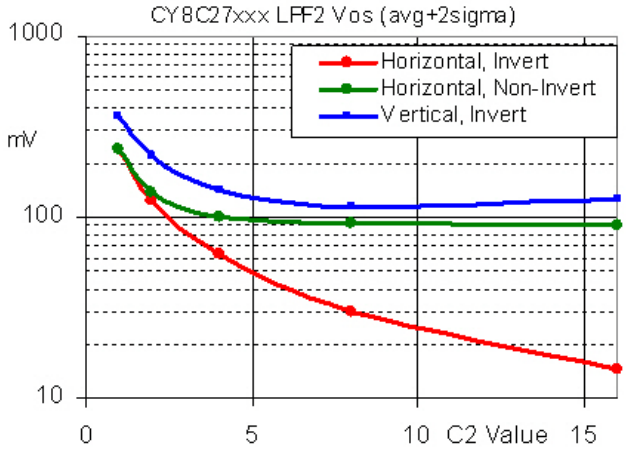


Figure 14. Vos vs Temp

TBD

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 two-pole low-pass filter has the transfer function:

Equation 1

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

where: Gain is the DC gain of the filter
d is the damping factor
ω_0 is the natural frequency
ω_n is the normalized corner frequency
s is the Laplace operator = 2*π*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 roll-off of 12 dB for each octave at frequencies up to half of the sample rate. The recommended OSR is between 6 and 100.

The filters can be cascaded to make more complex forms; see the LPF4 User Module datasheet and AN67391 for examples.

The pass-band performance of standard filter types for a nominal corner frequency of 10 kHz is shown in Figure 15. 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 16.

Figure 15. Low-pass Filter Pass-band Response

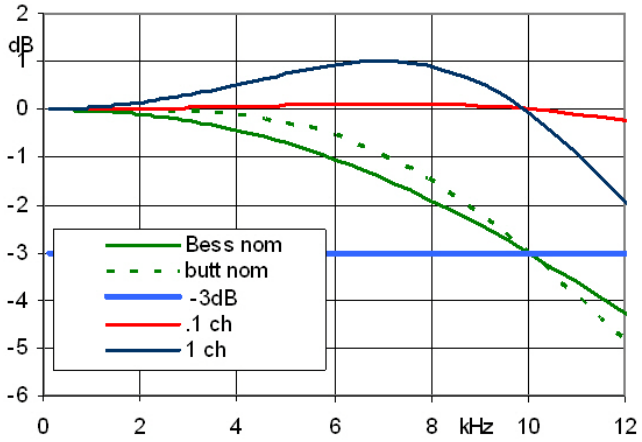
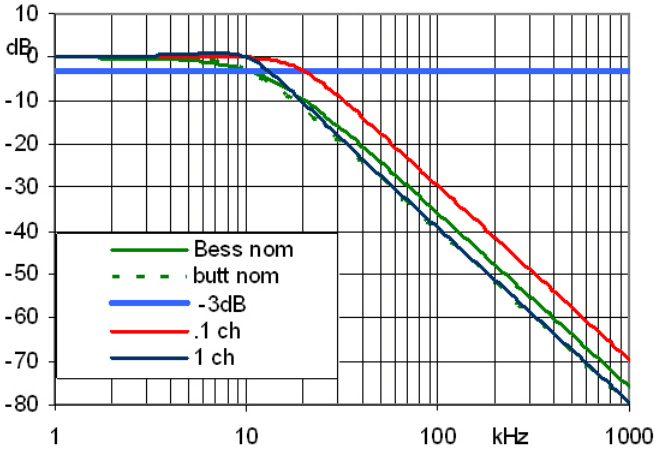
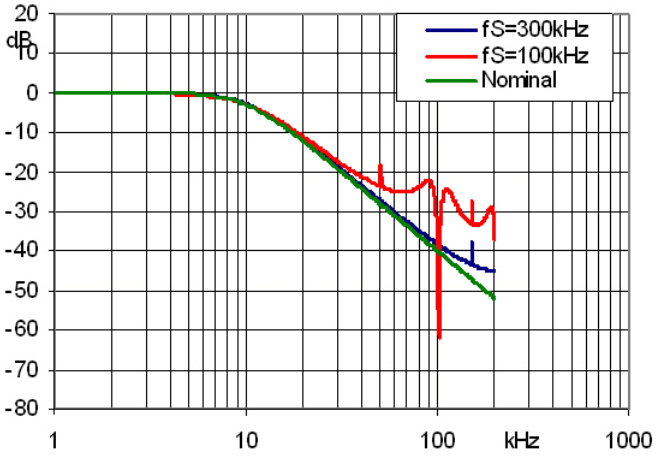


Figure 16. Low-pass Filter Response



The LPF2 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 17.

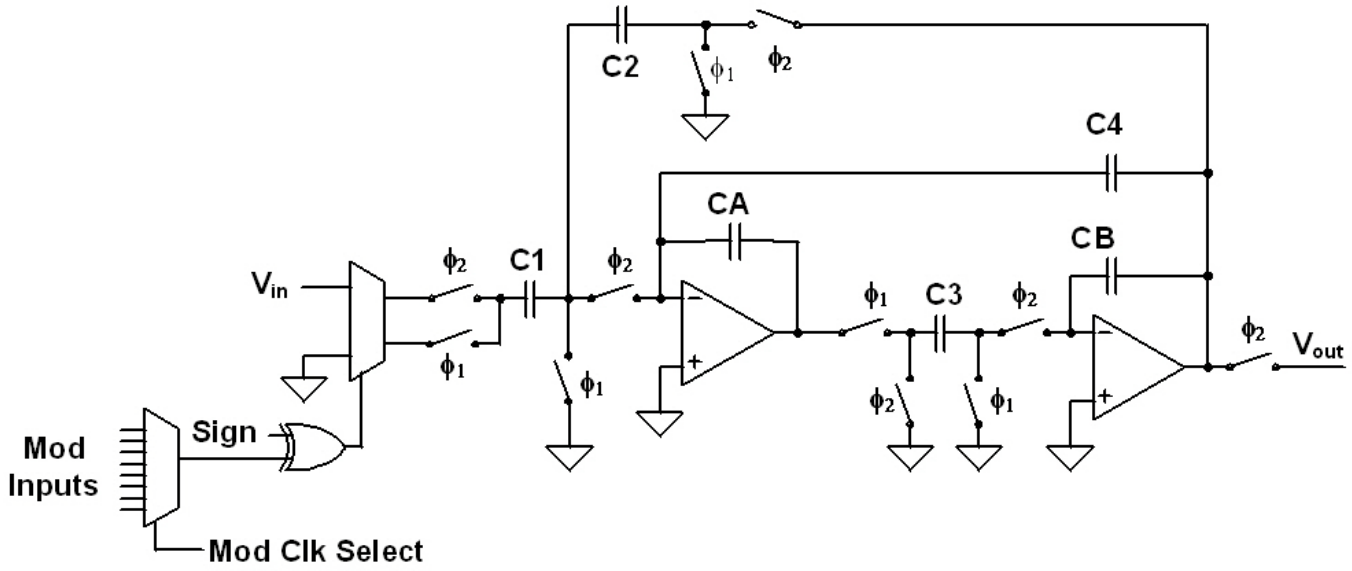
Figure 17. Switched Capacitor Filter Low-pass Response



Functional Description

An outline schematic of the LPF2 is shown in Figure 18. The filter uses two switched-capacitor blocks and one or two column clocks (at the same frequency) depending on placement. Polarity and modulation inputs are user-selectable.

Figure 18. 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 LPF2 is:

Equation 2

$$\frac{Vout}{Vin} = \frac{-\frac{C1}{C2}\frac{\left(1-\left(\frac{s}{2f_S}\right)^2\right)f_S{}^2}{\left(\frac{C_AC_B}{C_2C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}}{s^2+\frac{C_4}{C_2}\frac{sf_S}{\left(\frac{C_AC_B}{C_2C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}+\frac{f_S{}^2}{\left(\frac{C_AC_B}{C_2C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}}$$

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 2 to equation 1, yields a set of design equations:

Equation 3

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

Equation 4

$$(\omega_n\omega_0)^2 = \frac{f_S{}^2}{\left(\frac{C_AC_B}{C_2C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}$$

Equation 5

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

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 LPF2 is 180 kHz at high power and high bias, determined by opamp slewing and settling. The corner frequency has been tested down to 10 Hz at OSR = 100. The ultimate low corner frequency may be limited by noise at elevated temperature and aliasing of external signals. The user is advised to characterize these parameters in his own 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 2 for a unity gain filter when frequency (s=2 f) is very large, the attenuation 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. 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 LPF2 is set in the user's code using the APIs provided. The operational bandwidth of the switched-capacitor filter is determined by 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 V_{DD} = 5.0 V.

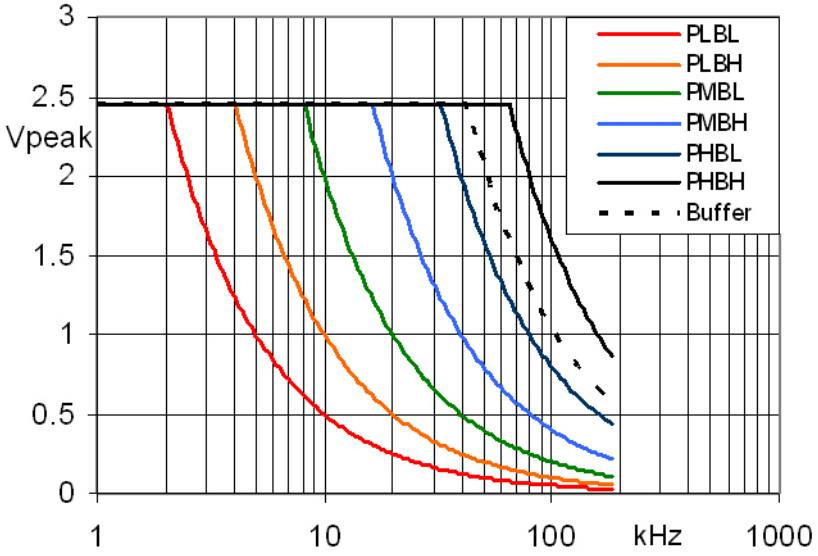
Table 5. Power Settings; Vdd = 5.0 V

	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

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 19. 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.

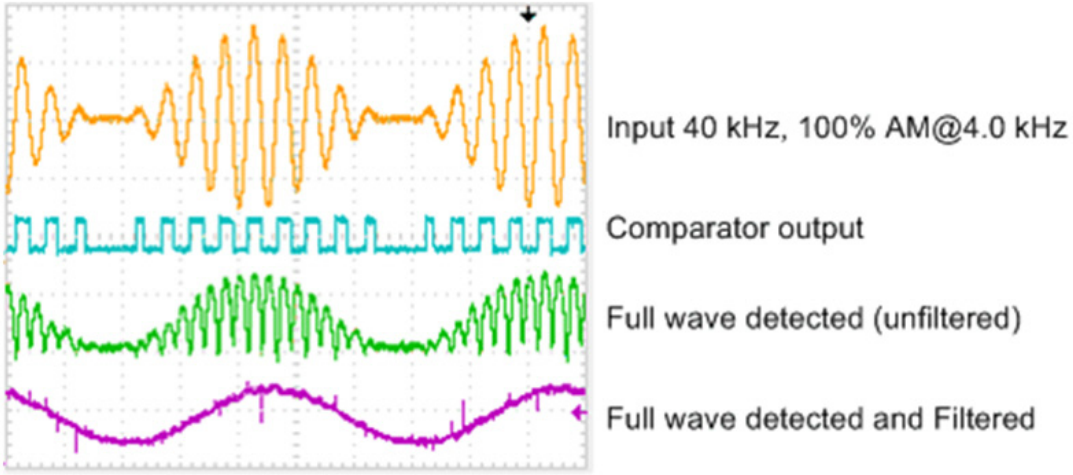
Polarity and Modulator Functions

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. In order 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.

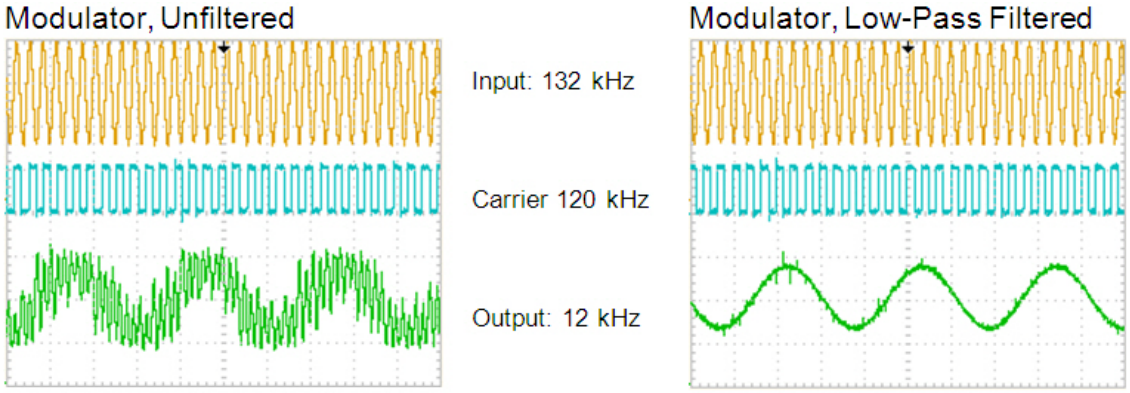
Another common usage of the modulator is a detector. The input fed to the modulator/filter is also fed to a comparator; the digital comparator output drives the modulator input of the filter. The result is full-wave detection. The example shows a 40 kHz input signal, 100% AM modulated at 4.0 kHz.

Figure 20. Full-Wave Detector Waveforms



The modulator can be used to mix frequencies up or down. The example in Figure 21 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 kHz.

Figure 21. 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 16 show 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 19. Higher power levels have higher integrated noise.

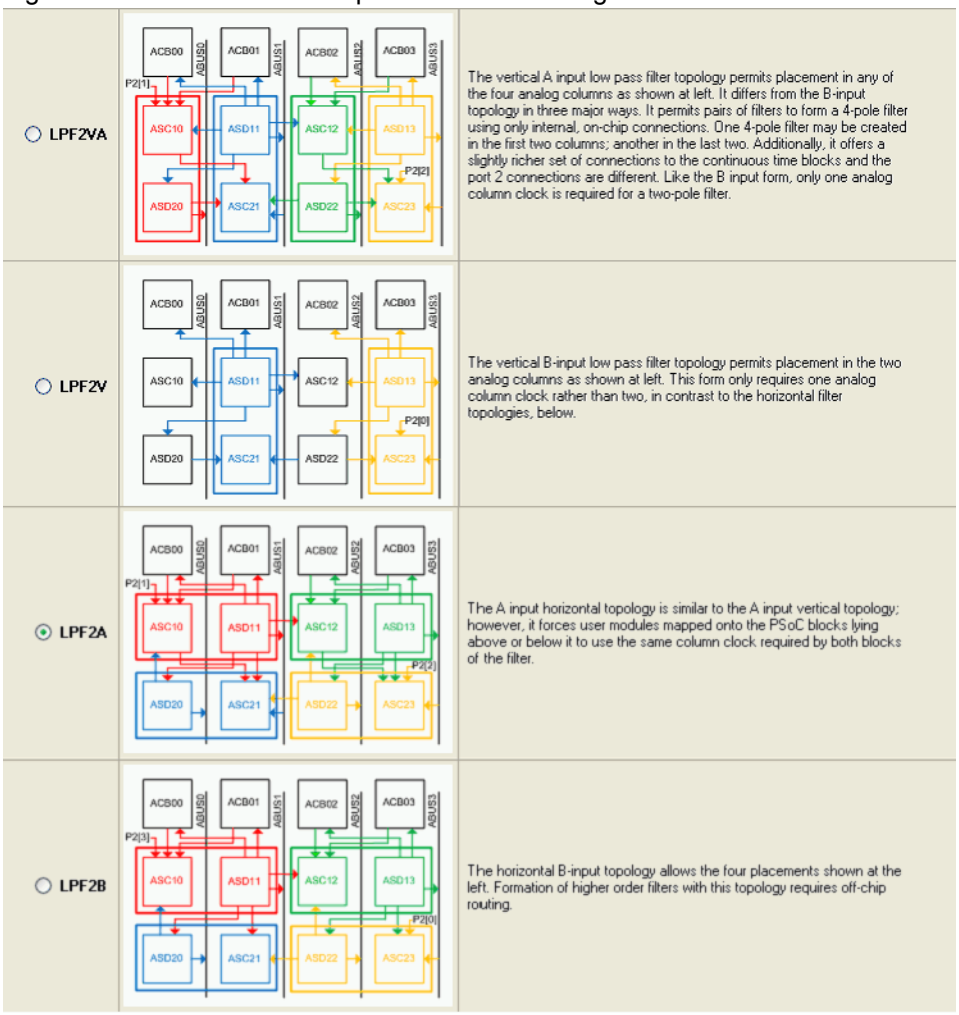
Anti-aliasing

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 LPF2'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 attenuates clock noise.

Placement

To implement a low-pass filter the user places an instance of the LPF2 in the analog array. A window opens showing the four variations in topology and their allowed placements. Inputs to the filter can connect to the A-Cap or B-Cap inputs of the switched-capacitor block; this allows a variety of non-overlapping input selections. Placements are either vertical (LPF2VA and LPF2VB) in a single column or horizontal (LPF2A and LPF2B) using two columns. When using horizontal topology filters, the columns must have the same clock. A Design Rule Check (DRC) at build verifies that this is correctly done. Filters using the modulator are allowed only when the input is on the A-Cap (LPF2A and LPF2VA). The filter may be dragged from one allowed location to the next. Changes in topology are accessed by right-click on the user module, then left-click on "Selection Options."

Figure 22. LPF2 Selection Options in PSoC Designer

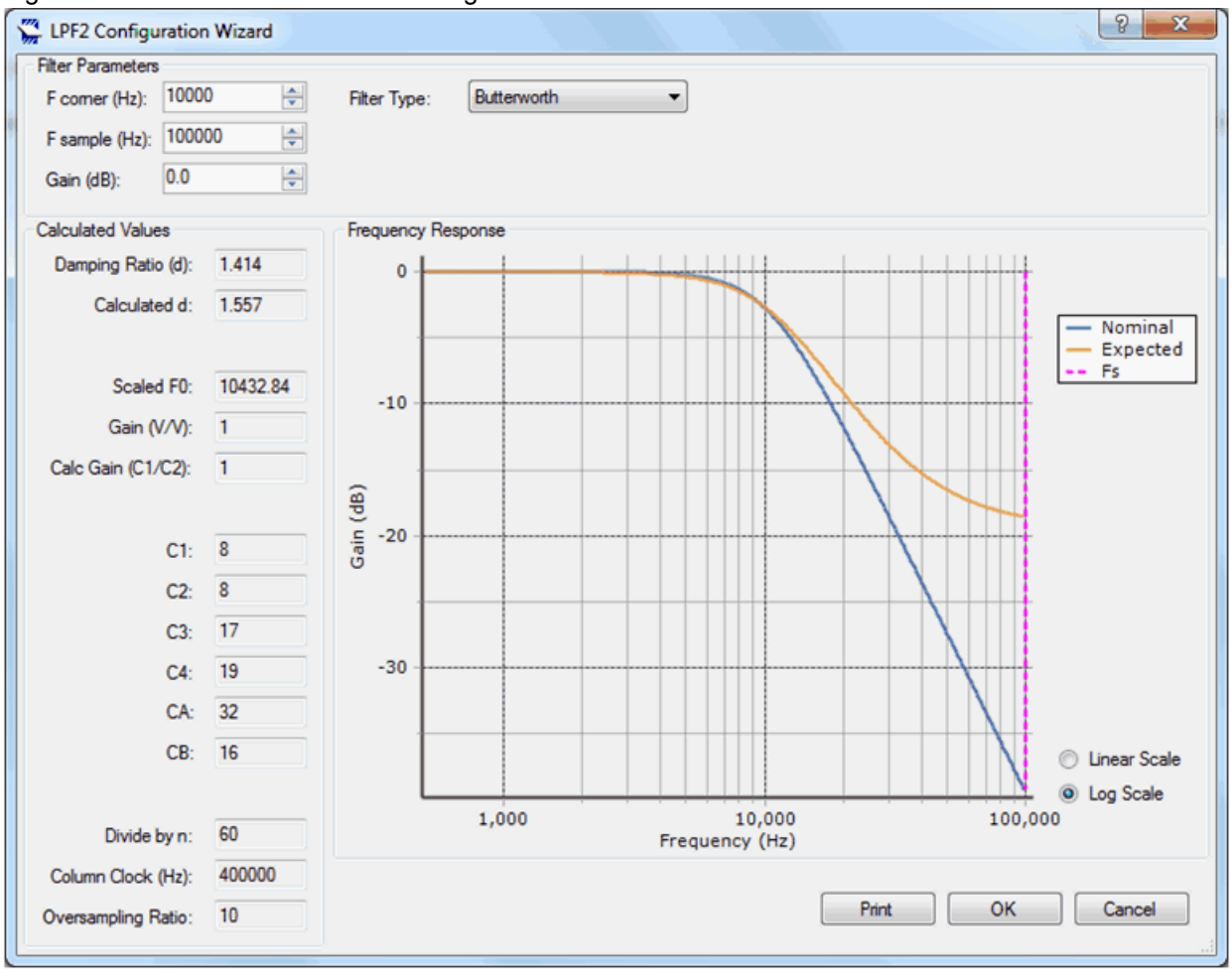


Wizard/Design

The design objective is usually to meet response requirements and achieve the highest possible sample rate for the best waveform fidelity and minimum aliasing of out-of-band signals. Other system requirements such as shared clocks may determine available sample rates.

After placing the filter in the design, the user right-clicks on the user module and chooses "Filter Design Wizard" from the menu. The user enters decimal values for gain, corner frequency, and sample rate parameters and selects the filter type from Bessel, Butterworth, 0.1 dB Chebychev, and 1.0 dB Chebychev in a pull-down list.

Figure 23. LPF2 Wizard in PSoC Designer



The nominal d and ω0 values are displayed along with the compensated values based on OSR. The wizard does not directly calculate the capacitor values. Instead, it calculates d and ω0 for all combinations of all capacitors and selects the set that gives the closest fit to the required compensated values. The OSR, required column clock, divider value, OSR and capacitor values are displayed. The plot shows lines for the nominal response of the filter, the expected response based on the available integer capacitor values and the locations of the sample rate and sample rate/2. When the user selects OK, the capacitor values are transferred to the parameter fields for the LPF2. The clock is separately set by the user and may be one of the available system clocks (VC1 or VC2) or from the output of one of the digital blocks. Note that for horizontally located filters, both columns must have the same clock source.

Capacitor values are integers. The calculation routine in the wizard finds the best fit, but there is no guarantee that it is a perfect fit, especially for high OSR (>60) which uses small capacitor values (≤ 4).

Most of the time, the d and ω0 values will be within a few percent of the design point and the filter response will track the nominal shape of the filter in the pass-band within 0.5 dB.

Parameters

Input

Inputs to the filter come from outputs of adjacent PSoC blocks, references and P2[0:3].

AnalogBus

The output of the filter can be connected to the analog output bus of the column containing the output block.

CompBus

The output block's comparator output can be connected to comparator bus. This enables connection to digital blocks or to an interrupt. Because the threshold for the internal comparator is fixed at AGND, the comparator functions as a zero-crossing detector.

Capacitor Values C1, C2, C3, C4, CA, and CB

The values of these capacitor values and the column clock determine the frequency response of the filter. C1, C2, C3, and C4 are integer values from 1 to 31. CA and CB are selectable as 16 or 32. You can design the capacitor values by the direct numerical approach in Appendix 2 or, preferably, using the design wizard that searches for the optimum set of capacitor values to meet your stated response requirements.

Polarity

This selects the polarity of the output with respect to the input. Select Inverting to set the output to invert the signal relative to AGND. When Gain is set greater than 1.0, the gain is relative to AGND. When Gain is set to 1.0 and Polarity is set to Non-Inverting, the signal is not referenced to AGND. The parameter applies only to topologies using the A-input.

Sample Frequency

The sample frequency is an essential part of the filter design but is not set in the LPF2 User Module.

The sample frequency is determined by the user to maintain the OSR in the range of 6 to 100. The wizard specifies the divider ratio from the system clock based on the sample frequency. The user is expected to instantiate a PWM or a counter at four times the sample frequency or use either VC1 or VC2 clocks.

Modulator Clock

The modulator uses the polarity control in the A input of the input block (FLIN). This function is available only for topologies LPF2A and LPF2VA. It is not available for LPF2B and LPF2V because these topologies use the B input of the input block, which does not have polarity control. There are eight sources (including OFF) for the modulator:

- None
- GlobalOutEven_0
- GlobalOutEven_1
- Row_0_Broadcast
- ComparatorBus_0
- ComparatorBus_1
- ComparatorBus_2

- ComparatorBus_3

  Mixers are constructed using clocks derived from the digital blocks routed through GlobaOutEven_0, GlobalOutEven_1, and Row_0_Broadcast. A full-wave detector is typically constructed by using the output of a PGA or BPF to drive the signal input of the filter, and using the output of a comparator or the comparator output of the BPF to drive the modulator input of the LPF. The full-wave detector works best when the BPF source and the LPF have the same clock frequency.

Applications Programming Interface (API)

The API routines are given as part of the user module to enable you to deal with the module at a higher level. Entry points are provided to initialize the LPF2 User Module, change power settings, and disable the user module. 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 before 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.

LPF2_Start

Description:

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

C Prototype:

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

Assembly:

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

Parameters:

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 listed in the following table:

Symbolic Name	Value
LPF2_OFF	0
LPF2_LOWPOWER	1
LPF2_MEDPOWER	2
LPF2_HIGHPOWER	3

Note For proper performance, filters with corner frequencies above 40 kHz should (1) use LPF2_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.

LPF2_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  LPF2_SetPower(BYTE bPowerSetting)
```

Assembly:

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

Parameters:

bPowerSetting: Same as the bPowerSetting used for the Start entry point.

Return Value:

None

Side Effects:

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

LPF2_SetCA, SetCB

Description:

Sets the value of the feedback capacitors in the user module FLIN block (CA) and FLOUT block (CB). This allows on-the-fly modification of the low-pass filter transfer function.

C Prototype:

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

Assembly:

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

Parameters:

FEEDBACK_CONSTANT: One byte that specifies the size of the feedback capacitors CA or CB (see the LPF2 Block Diagram). Symbolic names are given in the C and assembly include files; their associated values are listed in the following table:

Symbolic Name	Value
LPF2_FEEDBACK_16	0x00
LPF2_FEEDBACK_32	0x01

Return Values:

None

Side Effects:

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

LPF2_SetC1, SetC2, SetC3, and SetC4

Description:

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

C Prototype:

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

Assembly:

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

Parameters:

CapValue: Integer value from 1 to 31 for C1, C2, C3, and C4 (see the LPF2 Block Diagram). Values outside this range are truncated modulo 32.

Return Values:

None

Side Effects:

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

LPF2_SetPolarity (A-Input Topology Filters Only)

Description:

Sets the polarity of the output signal by selecting whether to invert or not to invert the input signal on FLIN. This allows on-the-fly modification of the low-pass filter output polarity. This function applies only to A-input topology filters.

C Prototype:

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

Assembly:

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

Parameters:

POLARITY_CONSTANT: One byte that specifies whether to invert or not to invert. Symbolic names are given in the C and assembly include files; their associated values are listed in the following table:

Symbolic Name	Value
LPF2_POLARITY_INVERTING	0x00
LPF2_POLARITY_NON_INVERTING	0x01

Return Values:

None

Side Effects:

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

LPF2_Stop

Description:

Powers off the user module.

C Prototype:

```
void  LPF2_Stop(void)
```

Assembly:

```
lcall  LPF2_Stop
```

Parameters:

None

Return Value:

None

Side Effects:

None.

Sample Firmware Source Code

The capacitor values for the filter are set by the user in the Wizard or by assigning values in the user module's parameter window. In C, using the low-pass filter is as simple as using the Start API to begin operation and calling the Stop API to complete operation.

```
//
//  This sample shows how to create a Low Pass Filter with corner frequency 1kHz.
//
// OVERVIEW:
//
//  The LPF2 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 LPF2VA MUM configuration of the LPF2 user module.
//   2. Place it onto ASC10 and ASD20 blocks.
//   3. Rename the User Module's instance name to LPF2.
//   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 LPF2's Input parameter to ACB00.
//   6. Set the LPF2's AnalogBus parameter to AnalogOutBus_0.
//   7. Set the LPF2's Polarity parameter to Non-Inverting.
//   8. Leave the rest of UM parameters by default.
//   9. Set the AnalogColumn_Clock_0 to VC2 (on the interconnect view)
//   10. Set the AnalogOutBuf_0 to Port_0_3 (on the interconnect view).
//   11. Place the PGA UM onto ACB00 block.
//   12. Rename User Module's instance name to PGA.
//   13. Set the PGA's Gain parameter to 1.000.
//   14. Set the PGA's Input parameter to AnalogColumn_InputMUX_0.
//   15. Set the PGA's Reference parameter to AGND.
//   16. Set the PGA's AnalogBus parameter to Disable.
//   17. 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 LPF2 and PGA.
//   2. The Analog Column clock should be 400 kHz to get the 1kHz low Pass Filter
//      with Over Sampling Ration = 100.
//
//
//   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
//  ------------------------------------------------------------------------------
```

```
//  LPF2        Name              LPF2                  UM's instance name
//              C1                1                     Set by Wizard
//              C2                1                     Set by Wizard
//              C3                4                     Set by Wizard
//              C4                22                    Set by Wizard
//              CA                32                    Set by Wizard
//              CB                32                    Set by Wizard
//              Input             ACB00
//              AnalogBus         AnalogOutBus_0
//              ÑompBus           Disable               Default value
//              Polarity          Non-Inverting
//              Modulator Clock   None                  Default value
//
// PGA          Name              PGA                   UM's instance name
//              Gain              1.000
//              Input             AnalogColumn_InputMUX_0
//              Reference         AGND
//              AnalogBus         Disable
//  ------------------------------------------------------------------------

/* Code begins here */

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

void main(void)
{
    // M8C_EnableGInt ;            // Uncomment this line to enable Global Interrupts
    PGA_Start(PGA_HIGHPOWER);      // Turn on the PGA
    LPF2_Start(LPF2_HIGHPOWER);    // Turn on the LPF
}
```

The equivalent assembly language is:

```
;
;  This sample shows how to create a Low Pass Filter with corner frequency 1kHz.
;
; OVERVIEW:
;
;  The LPF2 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 LPF2VA MUM configuration of the LPF2 user module.
;   2. Place it onto ASC10 and ASD20 blocks.
;   3. Rename the User Module's instance name to LPF2.
;   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 LPF2's Input parameter to ACB00.
;   6. Set the LPF2's AnalogBus parameter to AnalogOutBus_0.
```

```
;   7. Set the LPF2's Polarity parameter to Non-Inverting.
;   8. Leave the rest of UM parameters by default.
;   9. Set the AnalogColumn_Clock_0 to VC2 (on the interconnect view)
;   10. Set the AnalogOutBuf_0 to Port_0_3 (on the interconnect view).
;   11. Place the PGA UM onto ACB00 block.
;   12. Rename User Module's instance name to PGA.
;   13. Set the PGA's Gain parameter to 1.000.
;   14. Set the PGA's Input parameter to AnalogColumn_InputMUX_0.
;   15. Set the PGA's Reference parameter to AGND.
;   16. Set the PGA's AnalogBus parameter to Disable.
;   17. 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 LPF2 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
;  ------------------------------------------------------------------------------
;  LPF2       Name              LPF2                  UM's instance name
;             C1                1                     Set by Wizard
;             C2                1                     Set by Wizard
;             C3                4                     Set by Wizard
;             C4                22                    Set by Wizard
;             CA                32                    Set by Wizard
;             CB                32                    Set by Wizard
;             Input             ACB00
;             AnalogBus         AnalogOutBus_0
;             ÑompBus           Disable               Default value
;             Polarity          Non-Inverting
;             Modulator Clock   None                  Default value
;
; 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, LPF2_HIGHPOWER
    lcall LPF2_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. After the transfer function is chosen, the LPF2_SetC1 API function may be used to implement a programmable-gain control.

Appendix 1: Registers

The topology and placement of the LPF2 User Module determine half of the bits in the configuration registers for the blocks used. Bit fields that are independent of placement 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 variable bitfields used in the register definitions follow at the end of this appendix. Further details of registers may be found in the Technical Reference Manual.

Horizontal A-Input Topology

Table 6. Block FLIN, ASCxxCRx

Bit	7	6	5	4	3	2	1	0
CR0	CA	0	Polarity	C1				
CR1	Input			C2				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	Feedback		Power	

AMD_CR0

Table 7. AMD_CR0 for modulator in Column 0 or 2

Bit	7	6	5	4	3	2	1	0
	0	AMOD select column 2			0	AMOD select column 0		

AMD_CR1

Table 8. AMD_CR1 for Modulator in Column 1 or 3

Bit	7	6	5	4	3	2	1	0
	0	AMOD select column 3			0	AMOD select column 1		

Horizontal B-Input Topology

Table 9. Block FLIN, ASCxxCRx

Bit	7	6	5	4	3	2	1	0
CR0	CA	0	1	C2				
CR1	Feedback			C1				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	Input		Power	

Table 10. Block FOUT, ASDxxCRx

Bit	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	Power	

Vertical A-Input Topology

Table 11. Block FLIN, ASCxxCRx

Bit	7	6	5	4	3	2	1	0
CR0	CA	0	Polarity	C1				
CR1	Input			C2				
CR2	0	0	0	C4				
CR3	0	0	1	0	Feedback		Power	

Table 12. Block FLOUT, ASDxxCRx

Bit	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	Power	

AMD_CR0

Table 13. AMD_CR0 for Modulator in Column 0 or 2

Bit	7	6	5	4	3	2	1	0
	0	AMOD select column 2			0	AMOD select column 0		

AMD_CR1

Table 14. AMD_CR1 for Modulator in Column 1 or 3

Bit	7	6	5	4	3	2	1	0
	0	AMOD select column 3			0	AMOD select column 1		

Vertical B-Input Topology

Table 15. Block FLIN, ASCxxCRx

Bit	7	6	5	4	3	2	1	0
CR0	CA	0	1	C2				
CR1	Feedback			C1				
CR2	0	0	0	0	0	0	0	0
CR3	0	0	1	0	Input		Power	

Table 16. Block FLOUT, ASDxxCRx

Bit	7	6	5	4	3	2	1	0
CR0	CB	0	Polarity	C3				
CR1	FBIN			0	0	0	0	1
CR2	AnalogBus	CompBus	0	C4				
CR3	0	0	1	0	0	1	Power	

Variable BitField Definitions

The following definitions apply to all preceding register definitions:

CA and **CB** set the FLIN and FLOUT feedback capacitors, respectively, to either 16 or 32 units (see the LPF2 Block Diagram). CA and CB are configured in the Design Wizard or directly in the Device Editor.

C1, **C2**, **C3,** and **C4** set the capacitors to integer values between 1 and 32. Values are calculated in the filter Design Wizard or are set directly in the Device Editor using values calculated using the direct design procedure in Appendix 2.

Input controls the multiplexer that selects the input signal.The user module "Input" parameter determines the value of this bitfield. The value of the Input parameter can be set using the routing features of the Device Editor or entered in the LPF2 User Module parameters window.

AnalogBus enables connection of the filter output to the analog bus. The value of the AnalogBus parameter is manually configured using the routing pull-downs in the Device Editor or entered in the parameters window.

CompBus enables connection of the filter output to the comparator bus. The value of the CompBus parameter is manually configured using the routing pull-downs in the Device Editor or entered in the parameters window.

Feedback is the C2 feedback connection, automatically determined by placement of the LPF2 User Module in the Device Editor.

FBIN is the connection from the FLIN output to the FLOUT input, automatically determined by placement of the LPF2 User Module in the Device Editor.

Polarity controls whether the output of the filter is inverted or non-inverted. This bit can be configured directly using Device Editor. This option is available only to A-input topology filters.

Power controls the On/Off state of the PSoC block and the operating current setting. It is set initially by calling the user module API function LPF2_Start and can be modified by calling the functions LPF2_SetPower and LPF2_Stop.

AMOD select column x enable source for analog modulator in appropriate column when modulator function is set.

Appendix 2: Direct Numerical Design

Direct calculation, derived from equations 3, 4, and 5 yields a set of capacitor values that work well for over-sample ratio (OSR) greater than 30. For lower OSR, the user is advised to use the calculation in the wizard, which uses a more complete optimization method.

Given f_{CORNER}, d, and OSR (f_{SAMPLE}/f_{CORNER}).

Calculate pre-warped corner frequency and damping

Equation 7

$$f_{PREWARP} = \frac{f_{CORNER} * OSR}{\pi} \tan\left(\frac{\pi}{OSR}\right)$$

Equation 8

$$d_{PREWARP} = d\left(1 + \left(\frac{\pi}{OSR}\right)^2\right)$$

Calculate C_2:

Equation 9

$$C_2 = \text{int}\left(\frac{2\pi(C_A C_B)^{1/2}}{d_{PREWARP} * OSR}\right)$$

Calculate C_4:

Equation 10

$$C_4 = \text{int}\left(\frac{C_2 d_{PREWARP}}{2\pi OSR}\right)$$

If $C_4 > 31$, decrement C_2, recalculate C_4:

Equation 11

$$C3 = \left(\frac{d_{PREWARP}^2 C_A C_B}{d_{PREWARP}^2 \left(\frac{C_2}{4} + \frac{C_4}{2}\right) + \frac{C_4}{C_2}}\right)$$

Calculate C_1 from Gain:

Equation 12

$$C_1 = Gain * C_2$$

Set clock frequency (PWM or clock resource) at f_{CORNER} * OSR * 4. If using a horizontal placement, verify that both columns have the same clock source. Set capacitor values and clock source in parameter windows, select input, connect output, enter start commands in code (C or assembler), build, and test.

Version History

Version	Originator	Description
2.5	DHA	Added Version History.
3.00	DHA	Redesigned the user module wizard and updated the user module datasheet.
4.00	SEG	Improved capacitor value calculation algorithm in user module wizard. Reworked user module datasheet.
4.10	MYKZ	1. Users can now store printer settings in User Module Wizard. 2. Corrected algorithm for calculating the wizard's chart data in the nominal series. 3. Improved algorithm of capacitor calculation 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.