Two-Pole Band-Pass Filter Datasheet BPF2V 6.00

Copyright © 2002-2013 Cypress Semiconductor Corporation. All Rights Reserved.

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, CY8CTST120, CY8CTMG120, CY8CTMA120, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, CY8C28x52						
	0	0	2	109	0	1

Features and Overview

- User programmable center frequency, Q, and Gain
- No external components required
- Center frequency 20 Hz to 200 kHz
- Automated design using wizard
- User-selected over-sample ratio (OSR), a ratio of sample frequency to corner frequency
- Built-in polarity control
- Built-in comparator output for use in full-wave detection and communication applications
- Built-in modulator for use in frequency translation and signal generation

The BPF2 User Module uses two switched-capacitor blocks to implement a general-purpose second order band-pass filter. Center frequency and Q are functions of the ratios of programmable on-chip capacitors and clock frequency; no external components are required. The user selects filter characteristics and clock frequency. Capacitor and clock divider values are automatically calculated in the design tool (wizard). BPF2 can be cascaded or combined with other filter types to make more complex filter structures.

The filter's output is biased at AGND, selected in the Global Resources.

DC and AC Electrical Characteristics

Table 1. DC Electrical Performance Characteristics, VDD = 5.0 V (Default filter for tests is 10.0 kHz, Q = 2, sample rate = 200 kHz (column clock = 800 kHz), gain = 1, modulation OFF, Power = High, Bias = High)

Symbol	Description	Conditions	Min	Typ	Max	Units
Vos	Input offset voltage	Default filter	–	–	80	mV
TCVos	Temp coefficient, input offset voltage	Default filter	–	–	50	µV/°C
Idd_LL	Operating current	Pwr = L, Bias = L	–	–	150	µA
Idd_LH		Pwr = L, Bias = H	–	–	260	µA
Idd_ML		Pwr = M, Bias = L	–	–	500	µA
Idd_MH		Pwr = M, Bias = H	–	–	1000	µA
Idd_HL		Pwr = H, Bias = L	–	–	2000	µA
Idd_HH		Pwr = H, Bias = H	–	–	4000	µA

Table 2. AC Electrical Performance Characteristics, VDD = 5.0 V (Default filter for tests is 10.0 kHz, Q = 2, sample rate = 200 kHz (column clock = 800 kHz), gain = 1, modulation OFF, Power = High, Bias = High)

Symbol	Description	Conditions	Min	Typ	Max	Units
f_{CENTER}	Center frequency	Min, OSR=100Max, OSR=6	0.01	–	200	kHz
f_{ERR100}	Center frequency error	Default filter	–	–	1.0	%[1]
Q_Err	Q (f_{CENTER} / -3dB BW) Error	Default filter	–	–	1.5	%
Vn_spect	In band noise at f_{CENTER}	Pwr = M, Bias = L	–	160	–	nV/rtHz
		Pwr = H, Bias = H	–	200	–	
Vn_int	Integrated noise f_{CENTER} /100 to f_{SAMPLE}	Pwr = M, Bias = L	–	3.0	–	mVrms
		Pwr = H, Bias = H	–	10.0	–	

Note 1. Scaled to 24.0 MHz clock, verify sysclk +/- 0.25%.

Typical Operating Characteristics

Figure 1. Q = 1, Response to fSAMPLE, OSR = 10

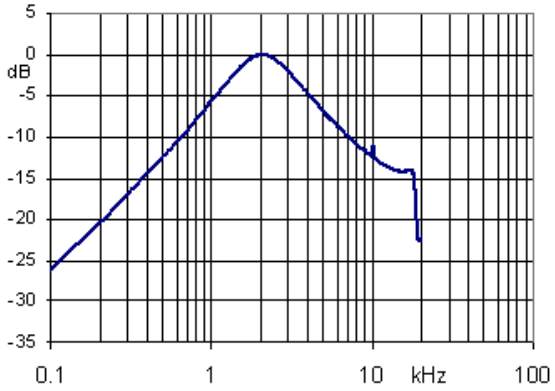


Figure 2. Q = 1, Response to fSAMPLE, OSR = 100

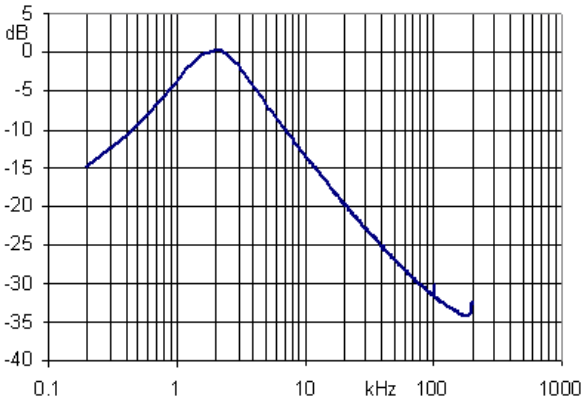


Figure 3. In-band Response, OSR = 100

TBD

Figure 4. In-band Response Q = 10 Response, OSR = 10

TBD

Figure 5. Q = 10, Response to fSAMPLE, OSR = 10

TBD

Figure 6. In-band Response Q = 10 Response, OSR = 100

TBD

Figure 7. Q = 10, Response to fSAMPLE, OSR = 100

TBD

Figure 8. Typical Noise Spectrum, 10 kHz Filter, OSR = 100

TBD

Figure 9. Typical Noise Spectrum,100 kHz Filter, OSR = 20

TBD

Figure 10. Noise Spectrum at Power Setting

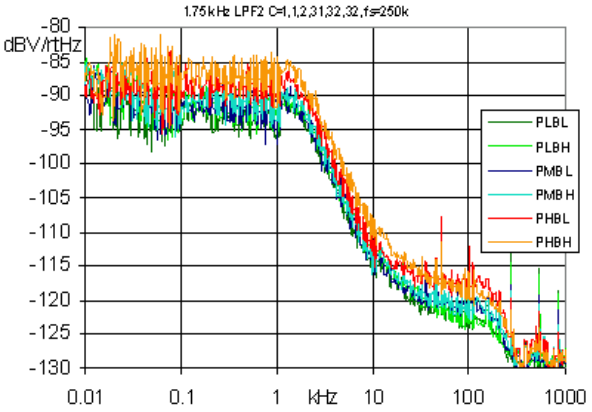


Figure 11. Integrated Noise at Power Setting

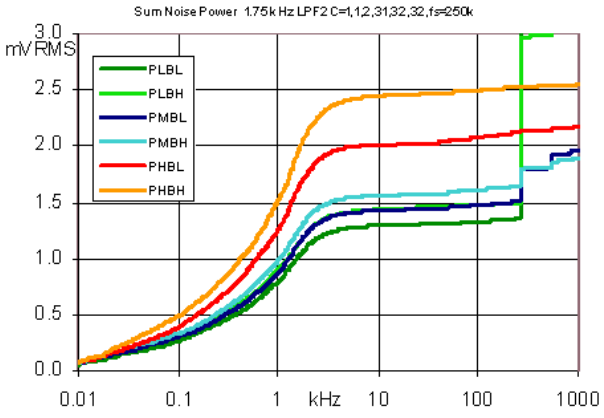


Figure 12. Typical Vos vs Cap Value

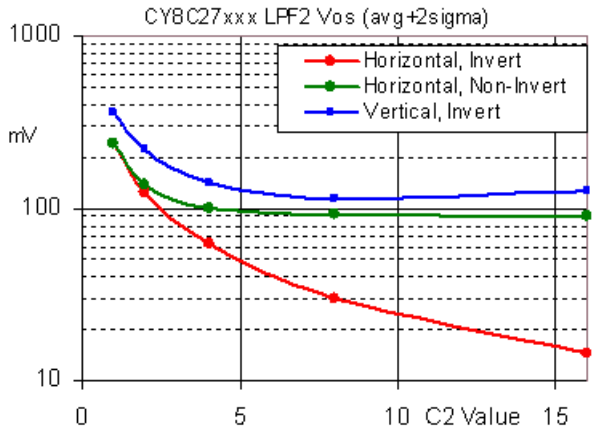


Figure 13. Vos vs Temp

TBD

Sufficient data shall be collected to provide graphical representation of maximum input offset voltage and TCVos versus C2 value for C2 = 1, 2, 4, 6, 8, 10, 12, 14 for filter configurations BPF2A, BPF2VA, BPF2B, BPF2V for both inverting and non-inverting polarities where available. For all filters, CA = 32, CB = 32.

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 band-pass filter has the transfer function:

Equation 1

$$\frac{V_{OUT}}{V_{IN}} = \frac{\frac{Gain}{Q} s\omega_0}{s^2 + \frac{s\omega_0}{Q} + \omega_0^2}$$

where Gain is the mid-band gain of the filter
Q is the quality factor (ratio of center frequency to –3dB bandwidth)
ω0 is the center frequency
s is the Laplace operator = j* ω.

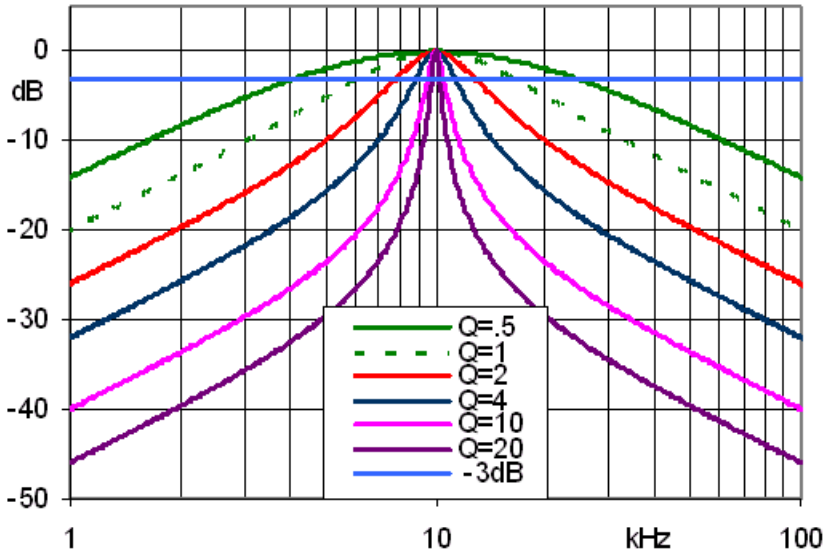
Filter performance is determined by the Q and center frequency. The center frequency is defined as the geometric mean of the upper and lower –3 dB frequencies:

Equation 2

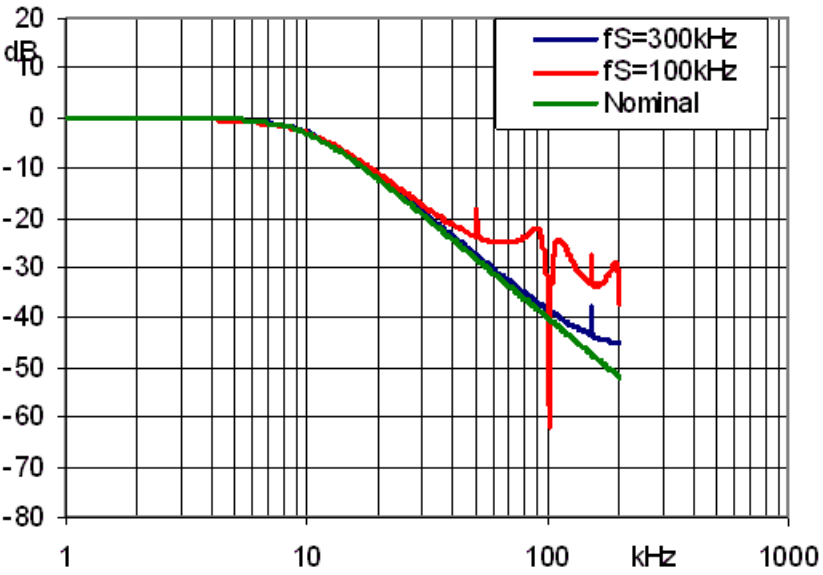
$$f_{CENTER} = \left(f_{-3dB_UPPER} * f_{-3dB_LOWER}\right)^{\frac{1}{2}}$$

Band-pass filters have a roll-off of 6 dB for each bandwidth octave at frequencies near the –3dB point, then at 6 dB for each absolute 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 BPF4 User Module Datasheet and AN67391 for examples.

Figure 14. Band-pass Filter Pass-band Response



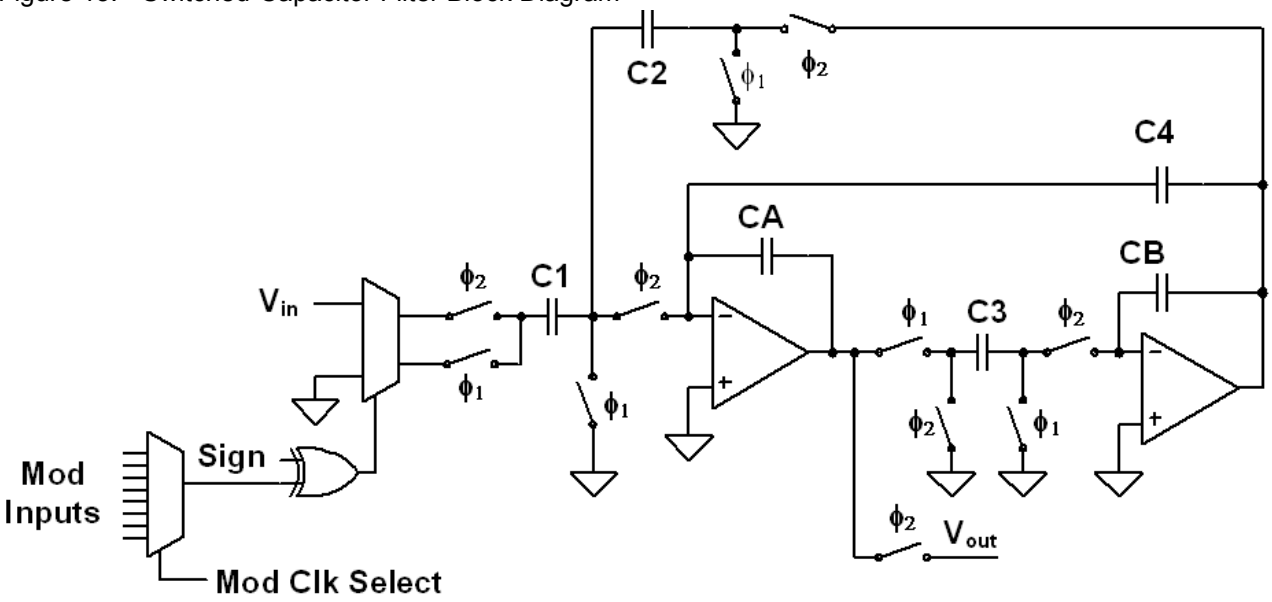
The BPF2 switched capacitor filter does not follow the ideal form of the band-pass response. The switching nature of the circuit results in a frequency-dependent gain in the numerator of the equation. This reduces the attenuation at higher frequencies, as shown by the example in Figure 3.



Functional Description

An outline schematic of the BPF2 User Module is shown in the following figure. 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. In the frequency domain, a single pole-pair band-pass filter has the frequency response shown in Equation 1:

Figure 15. 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 = jω) domain using the bi-linear transform. The transfer function of the BPF2 is:

Equation 3

$$\frac{Vout}{Vin} = \frac{-\frac{C_1 C_B}{C_2 C_3}\frac{\frac{s}{f_S}\left(1+\frac{s}{2f_S}\right)f_S^{\,2}}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}}{s^2+\frac{C_4}{C_2}\frac{sf_S}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}+\frac{f_S^{\,2}}{\left(\frac{C_A C_B}{C_2 C_3}-\frac{1}{4}-\frac{1}{2}\frac{C_4}{C_2}\right)}}$$

where fS is the sample rate (= column clock/4)
C1 through CB values are calculated and entered by user or calculated in the wizard and automatically entered.

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

Equation 4

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

Equation 5

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

Equation 6

$$Gain = \frac{C_1 C_B\left(1+\left(\pi\frac{f_{CENTER}}{f_{SAMPLE}}\right)^2\right)^{\frac{1}{2}}}{C_4 C_3}$$

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.

Operation Limits

The maximum corner frequency of the BPF2 is 200 kHz at high power and high bias, determined by opamp slewing and settling. The center frequency has been tested down to 10 Hz at OSR = 100. The ultimate low center 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 band-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 : $\frac{s}{f_S}\left(1+\frac{s}{2f_S}\right)$ 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 is:

Equation 7

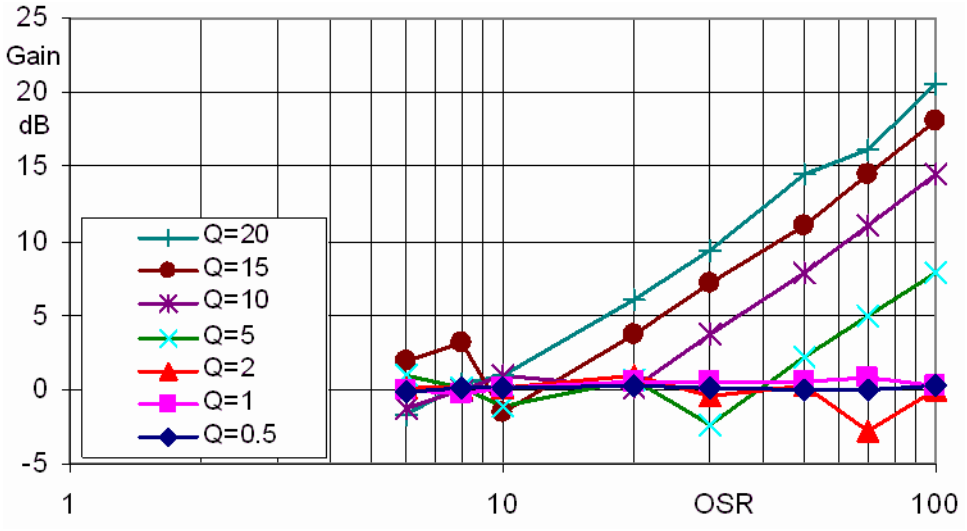
$$\frac{Vout}{Vin} = \frac{\frac{s}{f_S}\left(1+\frac{s}{2f_S}\right)\omega_0^2}{s^2} = \frac{(2\pi f_0)^2}{2f_S^2} = 2\left(\frac{\pi}{OSR}\right)^2$$

For OSR = 50 (2-kHz filter, 100 kHz sample rate), this is –42 dB at the sample frequency. The ultimate attenuation is independent of Q. The in-band performance of the filter is affected by the higher level

asymptote at low OSR, effectively increasing the frequency of the upper –3-dB point. The projected response of the wizard has been tested and demonstrated to agree closely with measured results for a wide range of frequency, Q, and OSR.

Filters with high OSR and moderate to high Q inherently have gain greater than unity because of the limitations of the available capacitor ratios. This is usually not usually a problem where small signals are amplified and filtered. When the input is large or the filter is used to create a sine wave from a square wave, the excess gain can be compensated by preceding the filter with a PGA configured for gain less than unity.

Figure 16. Filter Gain for Range of OSR and Q



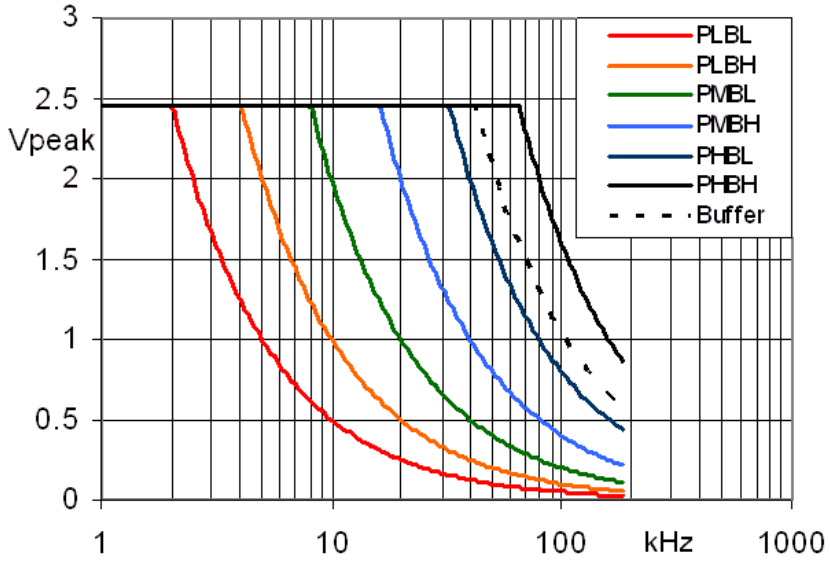
Power Setting

The operating current for the BPF2 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 VDD = 5.0 V.

	Power	Bias	Gain*Bandwidth (MHz)
PLBL	Low	Low	0.75
PLBH	Low	High	1.5
PMBL	Medium	Low	2.25
PMBH	Medium	High	3.1
PBL	High	Low	4.5
PHBH	High	High	9.0

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.

Figure 17. Maximum Signal vs Frequency



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

Polarity and Modulator Functions

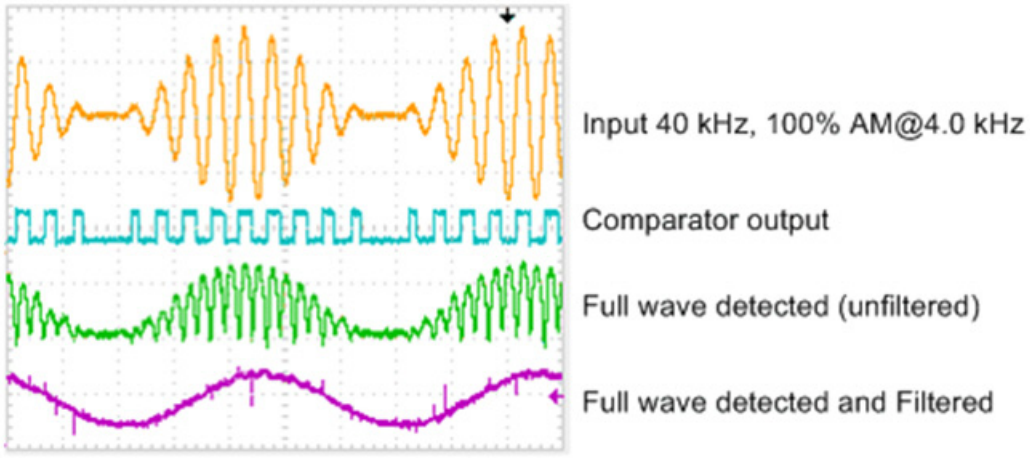
The input switched-capacitor block has a polarity control parameter. The polarity can also be controlled by a clock external to the block to form a modulator, multiplying the input by +1 or -1 at the clock rate. This generates signals at the sum of the carrier and input frequencies and at the difference 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 use 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 an envelope detector when combined with a low-pass filter. The analog output of the BPF2 is fed to the signal input of an LPF2. The comparator output of the BPF2 is

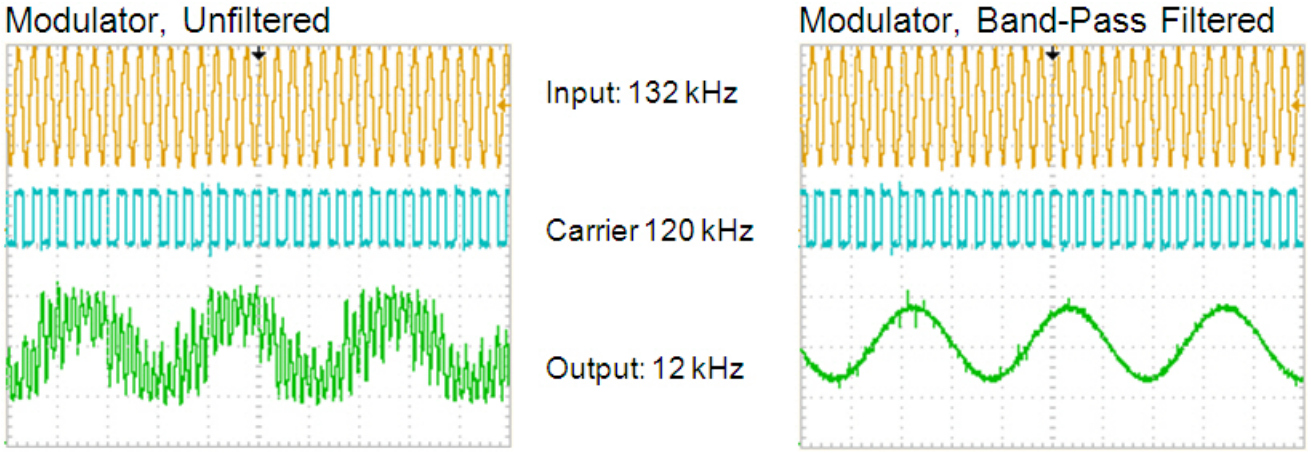
selected as the modulator source for the LPF2 modulator/filter. The result is full-wave detection. The example shows a 40 kHz input signal, 100% AM modulated at 4.0 kHz.

Figure 18. Full-Wave Detector Waveforms



The modulator can be used to mix frequencies up or down. The example in the following figure 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 19. 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. Higher power levels have higher integrated noise.

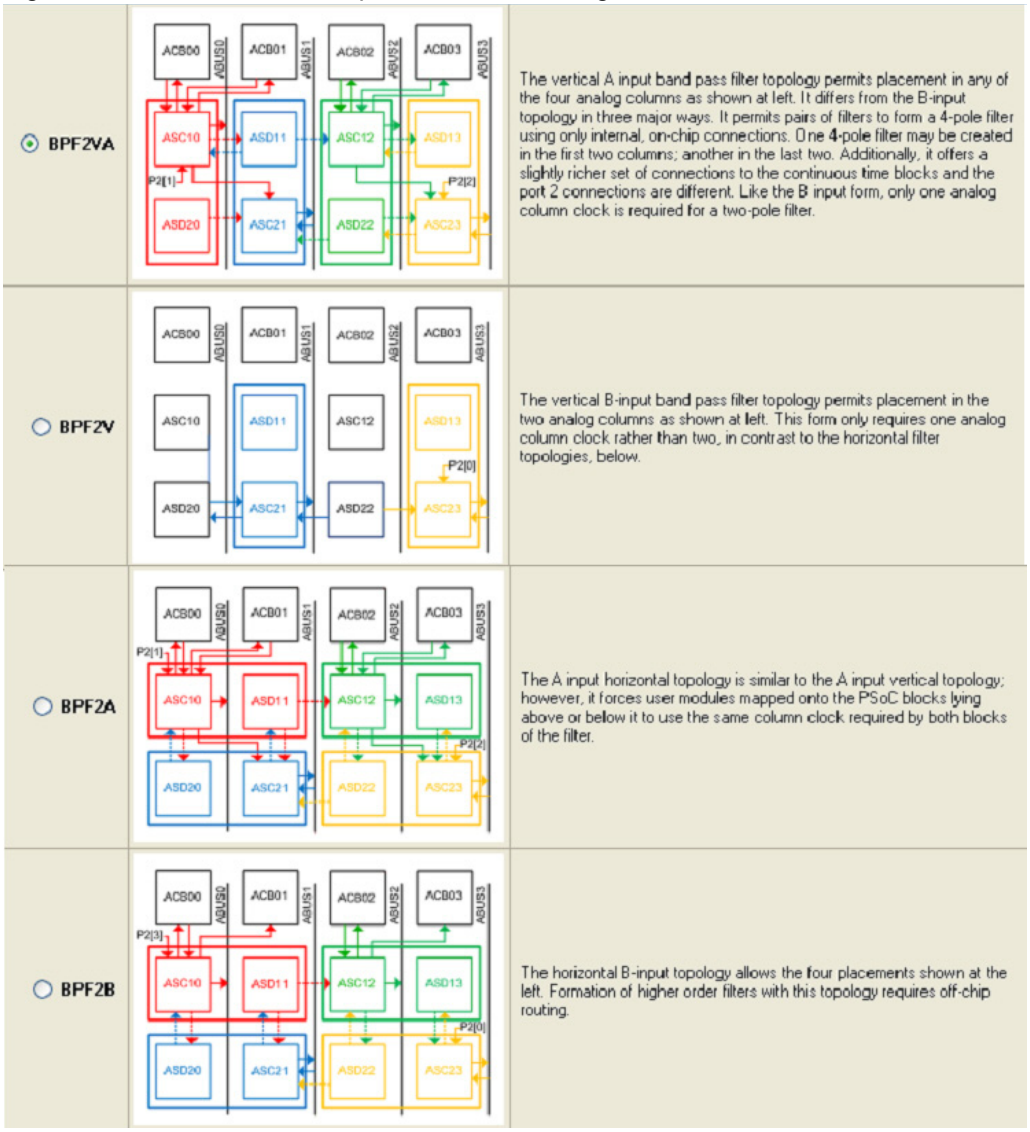
Anti-aliasing

The sampling nature of the switched-capacitor filters causes aliasing. This results in addition to the output of signals near multiples of the sample frequency. To provide optimum performance, the user should take care to limit the bandwidth of the input signal. This is typically done by using a single pole R-C filter with a corner frequency between the filters characteristic frequency and the sampling frequency.

Placement

To implement a band-pass filter, the user places an instance of the BPF2 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 (BPF2VA and BPF2VB) in a single column or horizontal (BPF2A and BPF2B) 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 (BPF2A and BPF2VA). The filter may be dragged from one allowed location to the next. You can change the topology by right-clicking the user module and then left-clicking **Selection Options**.

Figure 20. BPF2 Selection Options in PSoC Designer

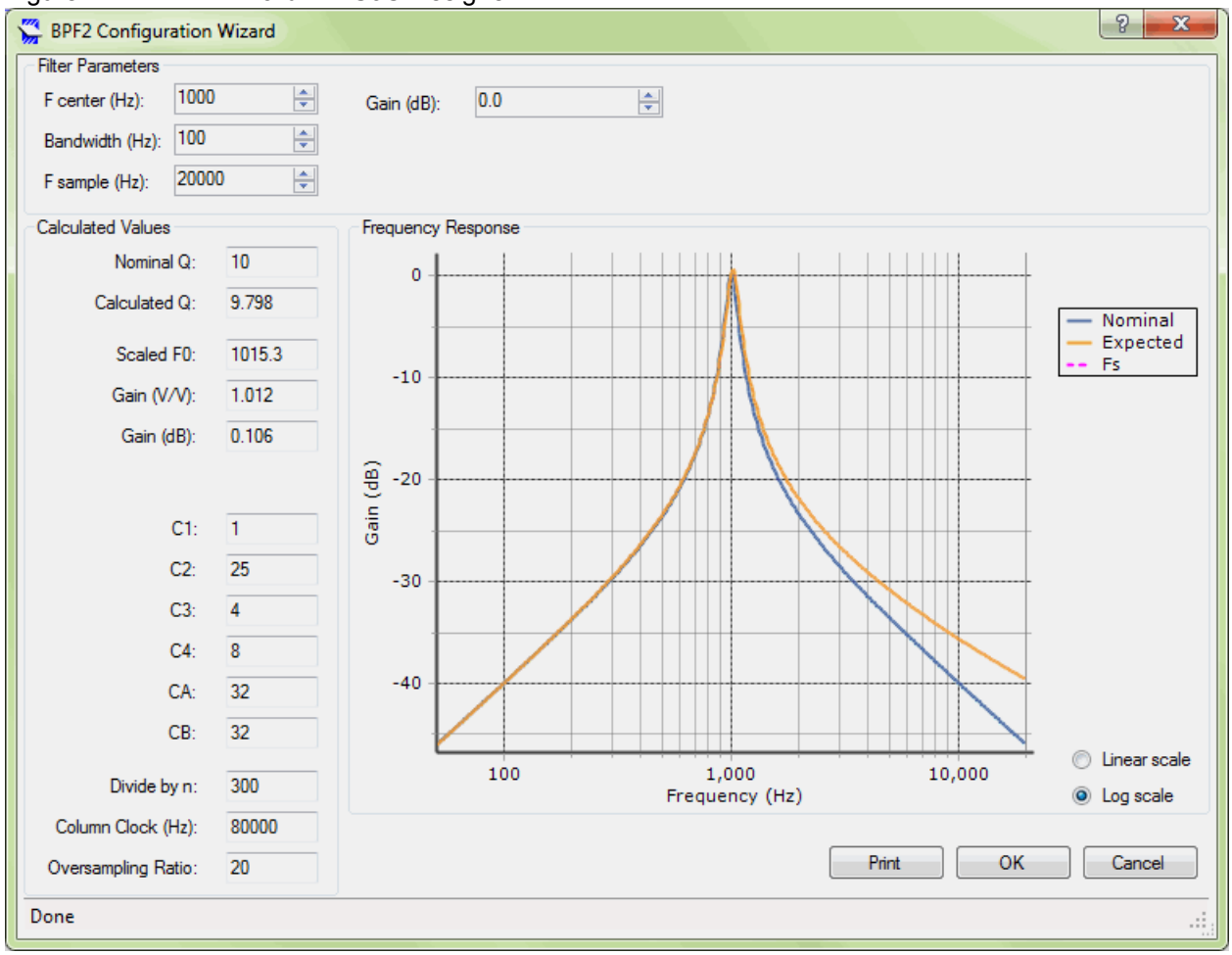


Wizard/Design

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, right-click the user module and choose "Filter Design Wizard" from the menu. Enter decimal values for gain, center frequency, bandwidth, and sample rate parameters.

Figure 21. BPF2 Wizard in PSoC Designer



The nominal Q and center frequency values are displayed along with the compensated values based on OSR. The wizard does not directly calculate the capacitor values. Instead, it calculates center frequency and Q for all combinations of capacitors and selects the set that gives the closest fit to the required compensated values. The OSR, required column clock, divider value, 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 sample rate. When the user selects OK, the capacitor values are transferred to the parameter fields for the BPF2. 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 Q and center frequency values are within a few percentage of the design point and the filter response tracks the nominal shape of the filter in the pass-band within 0.5 dB.

Parameters

To make a band-pass filter, place an instance of the BPF2 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 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. The threshold for the internal comparator is fixed at AGND, so 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 0 to 31. CA and CB are selectable as 16 or 32. Design of the capacitor values may be done 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 the user's stated response requirements.

Polarity

This parameter selects the polarity of the output with respect to the input. The output can be set to invert the signal relative to AGND by selecting Inverting. 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.

Modulator Clock

The modulator uses the polarity control in the A input of the input block (FLIN). This function is available only for topologies BPF2A and BPF2VA. It is not available for BPF2B and BPF2V because these topologies use the B input of the input block, which does not have polarity control. There are 8 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 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.

Sample Frequency

The sample frequency is an essential part of the filter design but is not set in the BPF2 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 counter at four times the sample frequency or use either VC1 or VC2 clocks.

Application Programming Interface

The Application Programming Interface routines are given as part of the user module to enable the user to deal with the module at a program rather than register level. Entry points are provided to initialize the BPF2 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 may 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.

BPF2_Start

Description:

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

C Prototype:

```
void  BPF2_Start(BYTE bPowerSetting)
```

Assembly:

```
mov   A, bPowerSetting
lcall  BPF2_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 are provided in the C and assembly include files; their associated values are listed in the following table:

Symbolic Name	Value
BPF2_OFF	0
BPF2_LOWPOWER	1
BPF2_MEDPOWER	2
BPF2_HIGHPOWER	3

Note For proper performance, filters with center frequencies above 40 kHz should use BPF2_HIGHPOWER and 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.

BPF2_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  BPF2_SetPower(BYTE bPowerSetting)
```

Assembly:

```
mov   A, bPowerSetting
lcall  BPF2_SetPower
```

Parameters:

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

Return Value:

None

Special Effects:

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

BPF2_SetCA, SetCB

Description:

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

C Prototype:

```
void  BPF2_SetCA(BYTE FEEDBACK_CONSTANT)
void  BPF2_SetCB(BYTE FEEDBACK_CONSTANT)
```

Assembly:

```
mov   A, FEEDBACK_CONSTANT
lcall  BPF2_SetCA                ; or, call  BPF2_SetCB
```

Parameters:

FEEDBACK_CONSTANT: One byte that specifies the size of the feedback capacitors CA or CB (see the BPF2 block diagram). Symbolic names are provided in the C and assembly include files; their associated values are listed in the following table:

Symbolic Name	Value
BPF2_FEEDBACK_16	0x00
BPF2_FEEDBACK_32	0x01

Return Values:

None

Side Effects:

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

BPF2_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 alteration of filter transfer characteristics by adjusting the other values.

C Prototype:

```
void  BPF2_SetC1(BYTE bCapValue)
void  BPF2_SetC2(BYTE bCapValue)
void  BPF2_SetC3(BYTE bCapValue)
void  BPF2_SetC4(BYTE bCapValue)
```

Assembly:

```
mov   A, CapValue
lcall  BPF2_SetC1               ; or, call  BPF2_SetC2 (or SetC3 or SetC4)
```

Parameters:

bCapValue: Integer value from 1 to 31 for C1, C2, C3 and C4 (see the BPF2 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.

BPF2_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 band-pass filter output polarity. This function applies only to the A-input topology filters.

C Prototype:

```
void  BPF2_SetPolarity(BYTE FEEDBACK_CONSTANT)
```

Assembly:

```
mov   A, FEEDBACK_CONSTANT
lcall  BPF2_SetPolarity
```

Parameters:

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

Symbolic Name	Value
BPF2_POLARITY_INVERTING	0x00
BPF2_POLARITY_NON_INVERTING	0x01

Return Values:

None

Side Effects:

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

BPF2_Stop

Description:

Powers the user module off.

C Prototype:

```
void  BPF2_Stop(void)
```

Assembly:

```
lcall  BPF2_Stop
```

Parameters:

None

Return Value:

None

Special Effects:

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

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, use Start API to begin operation and call the Stop API when done.

```
//
//  This sample shows how to create a Low Pass Filter with corner frequency 1kHz
//
// OVERVIEW:
//
//  The BPF2 input/output can be routed to any analog pin or adjacent analog block
// depending on placement.
//  In this example the BPF 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 BPF2VA MUM configuration of the BPF2 user module.
//   2. Place it onto ASC10 and ASD20 blocks.
//   3. Rename User Module's instance name to BPF2.
//   4. Run BPF2 Wizard from the context menu.
//      - Set the F center (Hz) parameter to 1000
//      - Set the Bandwidth (Hz) parameter to 100
//      - Set the F sample (Hz) parameter to 62500
//      - Leave the Gain parameter by default ("0.0")
//      - Click the "OK" button
//   5. Set BPF2's Input parameter to ACB00.
//   6. Set BPF2's AnalogBus parameter to AnalogOutBus_0.
//   7. Set BPF2's Polarity parameter to Non-Inverting.
//   8. Leave the rest of UM parameters by default.
//   9. Set AnalogColumn_Clock_0 to VC2 (on the interconnect view)
//   10. Set 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 PGA's Gain parameter to 1.000.
//   14. Set PGA's Input parameter to AnalogColumn_InputMUX_0.
//   15. Set PGA's Reference parameter to AGND.
//   16. Set PGA's AnalogBus parameter to Disable.
//   17. Set AnalogColumn_InputMux_0 to Port_0_5 (on the interconnect view)
//
// CONFIGURATION DETAILS:
//
//   1. The UM's instance names have to be shortened to BPF2 and PGA.
//   2. The Analog Column clock should be 250 kHz to get the 1kHz low Pass Filter with
// Over Sampling Ratio = 62.5.
//
// PROJECT SETTINGS:
//
//   1. Set the VC1=SysClk/N to 12
//   2. Set the VC2=VC1/N to 8
//   3. Set the A_Buff_Power to High
//
// USER MODULE PARAMETER SETTINGS:
//
//  ------------------------------------------------------------------------------
//  UM         Parameter         Value                 Comments
//  ------------------------------------------------------------------------------
```

```
//  BPF2        Name              BPF2                  UM's instance name
//              C1                1                     Set by Wizard
//              C2                5                     Set by Wizard
//              C3                2                     Set by Wizard
//              C4                16                    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
    BPF2_Start(BPF2_HIGHPOWER);   // Turn on the LPF
}
```

The equivalent assembly language code is:

```
;
;  This sample shows how to create a Low Pass Filter with corner frequency 1kHz.
;
; OVERVIEW:
;
;  The BPF2 input/output can be routed to any analog pin or adjacent analog block
;  depending on placement.
;  In this example the BPF 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 BPF2VA MUM configuration of the BPF2 user module.
;   2. Place it onto ASC10 and ASD20 blocks.
;   3. Rename User Module's instance name to BPF2.
;   4. Run BPF2 Wizard from the context menu.
;      - Set the F center (Hz) parameter to 1000
;      - Set the Bandwidth (Hz) parameter to 100
;      - Set the F sample (Hz) parameter to 62500
;      - Leave the Gain parameter by default ("0.0")
;      - Click the "OK" button
```

```
;   5. Set BPF2's Input parameter to ACB00.
;   6. Set BPF2's AnalogBus parameter to AnalogOutBus_0.
;   7. Set BPF2's Polarity parameter to Non-Inverting.
;   8. Leave the rest of UM parameters by default.
;   9. Set AnalogColumn_Clock_0 to VC2 (on the interconnect view)
;   10. Set 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 PGA's Gain parameter to 1.000.
;   14. Set PGA's Input parameter to AnalogColumn_InputMUX_0.
;   15. Set PGA's Reference parameter to AGND.
;   16. Set PGA's AnalogBus parameter to Disable.
;   17. Set AnalogColumn_InputMux_0 to Port_0_5 (on the interconnect view)
;
; CONFIGURATION DETAILS:
;
;   1. The UM's instance names have to be shortened to BPF2 and PGA.
;   2. The Analog Column clock should be 250 kHz to get the 1kHz low Pass Filter with
; Over Sampling Ratio = 62.5.
;
; PROJECT SETTINGS:
;
;   1. Set the VC1=SysClk/N to 12
;   2. Set the VC2=VC1/N to 8
;   3. Set the A_Buff_Power to High
;
; USER MODULE PARAMETER SETTINGS:
;
;  ------------------------------------------------------------------------------
;  UM         Parameter         Value                Comments
;  ------------------------------------------------------------------------------
;  BPF2       Name              BPF2                 UM's instance name
;             C1                1                    Set by Wizard
;             C2                5                    Set by Wizard
;             C3                2                    Set by Wizard
;             C4                16                   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, BPF2_HIGHPOWER
        lcall BPF2_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 BPF2_SetC1 API function may be used to implement a programmable-gain control.

Appendix 1: Registers

The topology and placement of the BPF2 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. Most of the variable bitfields 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. For further details of registers, refer to the Technical Reference Manual.

Horizontal A-Input Topology

Table 3. Block FLIN, ASCxxCRx

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

Table 4. Block FB, ASDxxCRx

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

Table 5. AMD_CR0 (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		

Table 6. AMD_CR1 (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 7. Block FLIN, ASCxxCRx

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

Table 8. Block FB, ASDxxCRx

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

Vertical A-Input Topology

Table 9. Block FLIN, ASCxxCRx

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

Table 10. Block FB, ASDxxCRx

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

Table 11. AMD_CR0 (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		

Table 12. AMD_CR1 (AMD_CR0 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 13. Block FLIN, ASCxxCRx

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

Table 14. Block FB, ASDxxCRx

Bit	7	6	5	4	3	2	1	0
CR0	CB	0	Polarity	C3				
CR1	FBIN			0	0	0	0	1
CR2	0	0	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:

CA and **CB** set the FLIN and FLFB feedback capacitors, respectively, to either 16 or 32 units (see the BPF2 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 value of the Input parameter can be set using the routing features of the Device Editor or entered in the BPF2 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 BPF2 User Module in the Device Editor.

FBIN is the connection from the FLIN output to the FLFB input, automatically determined by placement of the BPF2 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 BPF2_Start and can be modified by calling the functions BPF2_SetPower and BPF2_Stop.

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

Appendix 2: Direct Numerical Design

Direct calculation, derived from equations 4, 5, and 6 yields a set of "reasonable" capacitor values. The user is advised to use the calculation in the wizard, which uses a more complete, and easier, optimization method.

Given f_{CORNER}, f_{SAMPLE}, and Q

Calculate OSR = (f_{SAMPLE}/f_{CORNER})

Calculate pre-warped corner frequency and damping

Equation 8

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

Select provisional value for C2:

Equation 9

$$C_2 = Q$$

Caluclate C3:

Equation 10

$$C3 = \left(\frac{C_A C_B / C_2}{\frac{f_{SAMPLE}}{4\pi Q f_{PREWARP}} + \frac{1}{4} + \frac{f_{SAMPLE}}{2\pi^2 f_{PREWARP}}} \right)$$

Calculate C4:

Equation 11

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

Adjust C2 such that C3 and C4 are close to an integer value. Non near-integer values may significantly displace realized center frequency from the desired result.

Calculate C1 from gain:

Equation 12

$$C_1 = \frac{C_4 C_3}{C_B\left(1 + \left(\pi \frac{f_{CENTER}}{f_{SAMPLE}}\right)^2\right)^{\frac{1}{2}}}$$

Set clock frequency (PWM or clock resource) at f_{CORNER} * OSR * 4. If you use 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
5.5	SEG	Added Polarity and Modulator Functions sections and Power Setting for Application section.
5.60	DHA	1. Added help file to wizard. 2. Updated image and description of the BPF2 UM Filter Design Wizard in this user module datasheet.
6.00	SEG	1. Improved capacitor value calculation algorithm in user module wizard. 2. Reworked user module datasheet.
6.00.b	MYKZ	Users can now store printer settings in the 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 © 2002-2013 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.