Four-Pole Band-pass Filter Datasheet BPF4 V 2.10

Copyright © 2009-2013 Cypress Semiconductor Corporation. All Rights Reserved.

Resources	PSoC® Blocks			API Memory (Bytes)		Pins
	Digital	Analog CT	Analog SC	Flash	RAM	
CY8C29/28/24xxx, CY8C27x43						
	0	0	4	0	229	1[a]

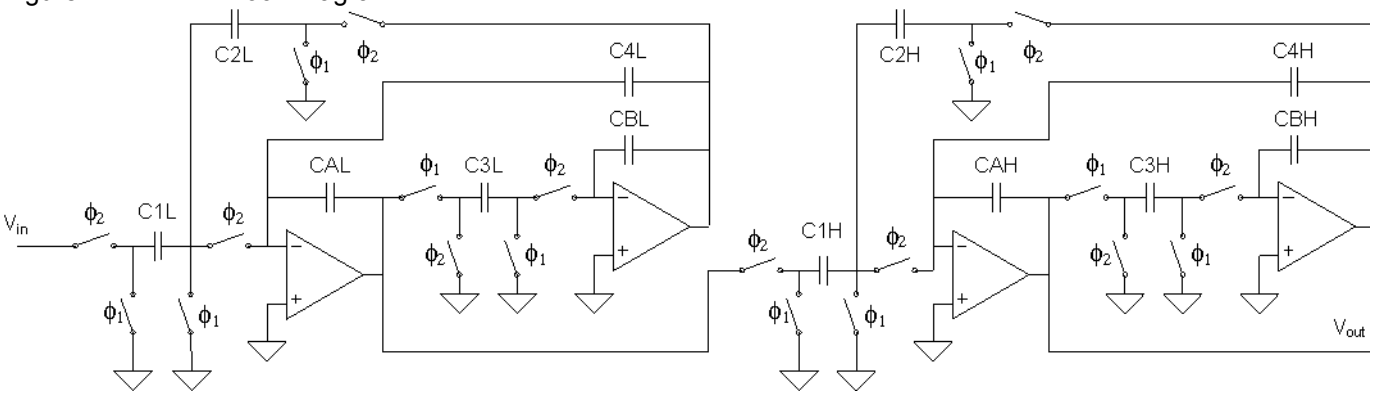
a. This number depends on the user module placement and “Input” user property selection.

Features and Overview

- User-programmable, mid-band gain
- User-programmable center frequency (20 Hz to 150 kHz) and Q with no external components
- Filter center frequency stability directly derived from clock accuracy
- Filter sampling rates up to 1.5 MHz
- Built in, zero-crossing detector

The BPF4 User Module implements a four-pole band-pass filter. The center frequency and Q (ratio of center frequency to bandwidth) are functions of the clock frequency and the ratios of the capacitor values chosen. The center frequency can be set accurately or adjusted by controlling the sample rate clock. Any of the classical all-pole filter configurations (Butterworth, Gaussian, Bessel, and Chebyshev) can be implemented. The filter output can drive the analog output bus.

Figure 1. BPF4 Block Diagram



Functional Description

In the frequency domain, a four-pole band-pass filter has the following frequency response:

Equation 1

$$\frac{V_{OUT}}{V_{IN}} = \frac{V_{OUT_L}}{V_{IN_L}} \times \frac{V_{OUT_H}}{V_{IN_H}}$$

The frequency response for Higher and Lower filter parts is calculated as:

Equation 2

$$\frac{V_{OUT_X}}{V_{IN_X}} = \frac{\frac{G_X}{Q_X}\omega_0 s}{s^2 + \frac{s\omega_0}{Q_X} + \omega_0^2}$$

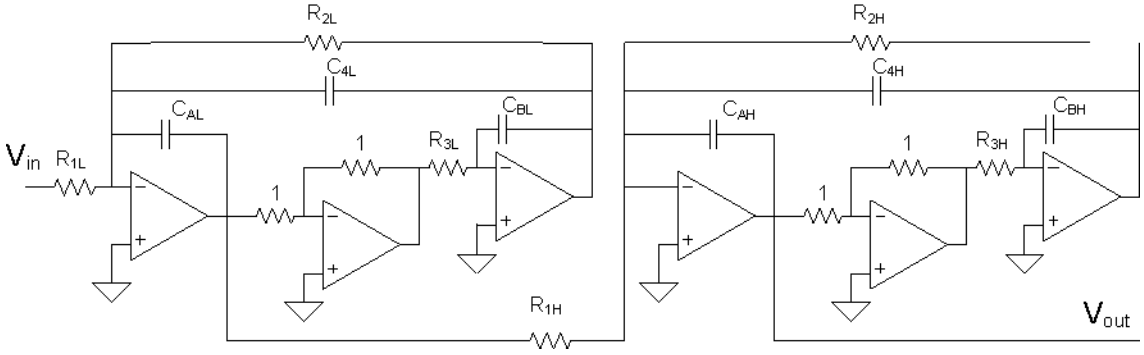
Note Depending on whether Lower or Higher filter parts are calculated, L or H indexes can be used in all formulas instead of X.

In Equation 2, Q is the ratio of center frequency to -3.0 dB bandwidth and ω0 is the center frequency. Multiple pole-pair band-pass filters have sections with identical Q and 0 scaled to meet transformed bandwidth requirements. All four-pole filters have far out-of-band attenuation asymptotic to 24 dB per octave (-12 dB per octave per pole). Band-pass filters have near out-of-band attenuation proportional to 24 dB per bandwidth octave. The initial out-of-band attenuation is steep. When the input frequency is more than twice the center frequency of the filter, then attenuation characteristic gradually becomes asymptotic to a slope of 24 dB per center frequency octave.

Required in-band performance and near-band attenuation requirements determine the type of band-pass filter chosen, for transformation into band-pass format. The standard Butterworth filter has monotonic amplitude performance and maximally flat phase shift in the pass band. Filters with low damping ratios (Chebyshev) have flatter in-band amplitude characteristics, but non-linear phase shift in the pass band and pulse response characterized by ringing. Filters with high damping ratios (Bessel) have linear phase shift in the pass band and pulse response characteristics with minimum over-shoot, but reduced near out-of-band attenuation. Values for band-pass poles are readily available in any filter design reference.

The basic form of the biquad filter is a pair of integrators with controlled DC and frequency dependent feedback paths. The biquad can be easily understood by examining the standard RC form of the biquad, shown in Figure 2:

Figure 2. RC-Biquad Block Diagram



The typical RC-biquad band-pass uses 3 opamps. The transfer function of the each RC-biquad is shown in Equation 3:

Equation 3

$$\frac{V_{OUT_X}}{V_{IN_X}} = \frac{-\frac{R_{2_X}}{R_{1_X}}\frac{1}{R_{2_X}C_{A_X}}s}{s^2 + s\frac{C_{4_X}}{R_{3_X}C_{A_X}C_{B_X}} + \frac{1}{R_{2_X}R_{3_X}C_{A_X}C_{B_X}}}$$

The transfer function of four-pole filter is equal to the multiplication of two RC-biquad transfer functions from what the four-pole has created:

Equation 4

$$\frac{V_{OUT}}{V_{IN}} = \frac{V_{OUT_L}}{V_{IN_L}} \times \frac{V_{OUT_H}}{V_{IN_H}}$$

In the PSoC switched capacitor implementation, the center inverting opamp is eliminated by reversing the polarity of the gain of the output block. Resistors are transformed into the switched capacitors, as shown in Figure 1.

Because of the nature of switched capacitor circuits as time-sampled devices, the transfer function is developed in the time domain, where z-1 is the time delay of one sample period, rather than the frequency domain (s=jω). The transfer function is converted to the frequency domain using the bilinear transform. The transfer function resolves to Equation 5:

Equation 5

$$\frac{V_{OUT_X}}{V_{IN_X}} = \frac{-\frac{C_{1_X}}{C_{2_X}}\frac{C_{B_X}}{C_{3_X}}\frac{s\left(1+\frac{s}{2f_s}\right)f_s}{\frac{C_{A_X}C_{B_X}}{C_{2_X}C_{3_X}} - \frac{1}{4} - \frac{1}{2}\frac{C_{4_X}}{C_{2_X}}}}{s^2 + \frac{C_{4_X}}{C_{2_X}}\frac{sf_s}{\left(\frac{C_{A_X}C_{B_X}}{C_{2_X}C_{3_X}} - \frac{1}{4} - \frac{1}{2}\frac{C_{4_X}}{C_{2_X}}\right)} + \frac{f_s^{\,2}}{\left(\frac{C_{A_X}C_{B_X}}{C_{2_X}C_{3_X}} - \frac{1}{4} - \frac{1}{2}\frac{C_{4_X}}{C_{2_X}}\right)}}$$

Comparing this equation with the standard form of Equation 1 yields a set of the design equations for Q, center frequency, fc, and Gain, G, as shown in the following equations:

Equation 6

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

Equation 7

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

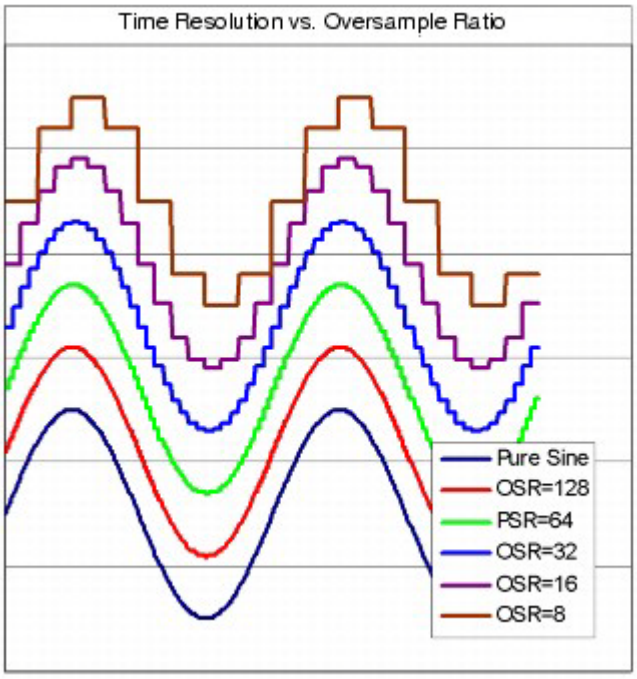
Equation 8

$$G_X = -\frac{C_{1_X}}{C_{4_X}}\frac{C_{B_X}}{C_{3_X}}$$

The numerator of Equation 5 has the term $1 + \frac{s}{2f_s}$, which results in reduced filter attenuation as the signal frequency approaches half of the Nyquist rate (that is, the sampling rate fs and a small amount of amplitude bias or upwards "tilt" in the pass band). This tilt is greater for lower Q and lower sampling frequencies. It can be compensated by adjusting the nominal bandwidth of the filter.

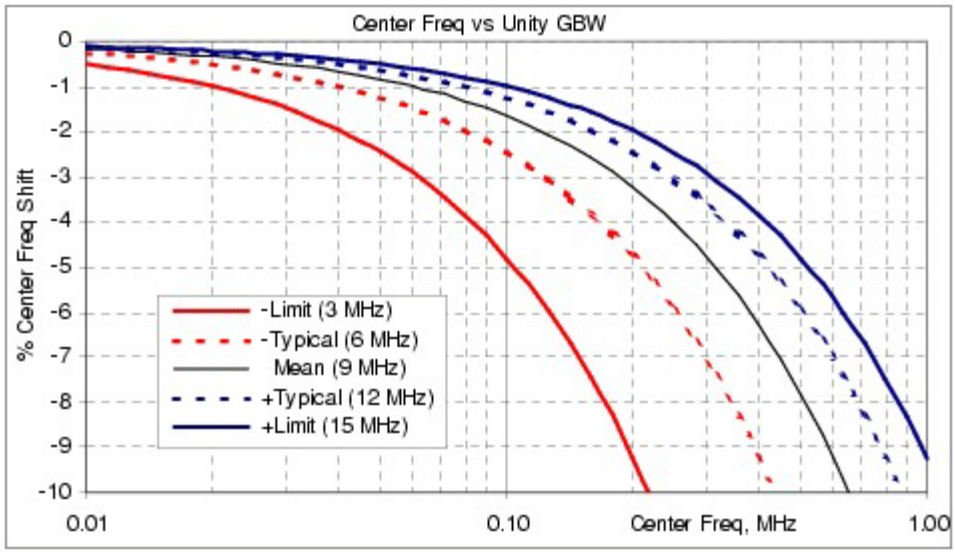
Higher sample rates result in filter performance closer to the standard form of Equation 1 and smoother waveforms as shown in Figure 3. This is true of all switched capacitor circuits, including filters, amplifiers, and DACs.

Figure 3. Time Resolution Vs. Oversample Ratio



The center frequency of the filter is controlled by the clock frequency, capacitor resolution, and PSoC opamp performance. At higher frequencies, the opamp's reduced open-loop gain becomes significant, with the center frequency being pushed low. The effect of the open-loop gain, as represented by the unity gain-bandwidth, on center frequency performance is shown in Figure 4. Filters with center frequencies above 40 kHz should have user module power set to “HIGHPOWER” and Opamp Bias set to “High” in the Global Parameters window.

Figure 4. Center Frequency Vs. Unity GBW



Power Setting for Applications

Switched capacitor block power and bias settings determine the performance of the filters. The power mode is selected using user module APIs. The bias is selected using the "Opamp Bias" parameter in the global resources window of PSoC Designer®. Power and bias settings determine the opamp operating current, which in turn determines the Slew Rate (SR). The opamp slew rate is 4 V/µsec when power and bias are both set High. It is reduced by ½ for each step down in power and bias.

The slew rate required to faithfully deliver the signal is dependent on peak voltage (V_{pk}) and frequency (F). The filter's output signal slews from its old value to its new value during one half of one phase of the clock. As a result, the filter requires an opamp slew rate four times the slew rate of the signal.

Equation 9

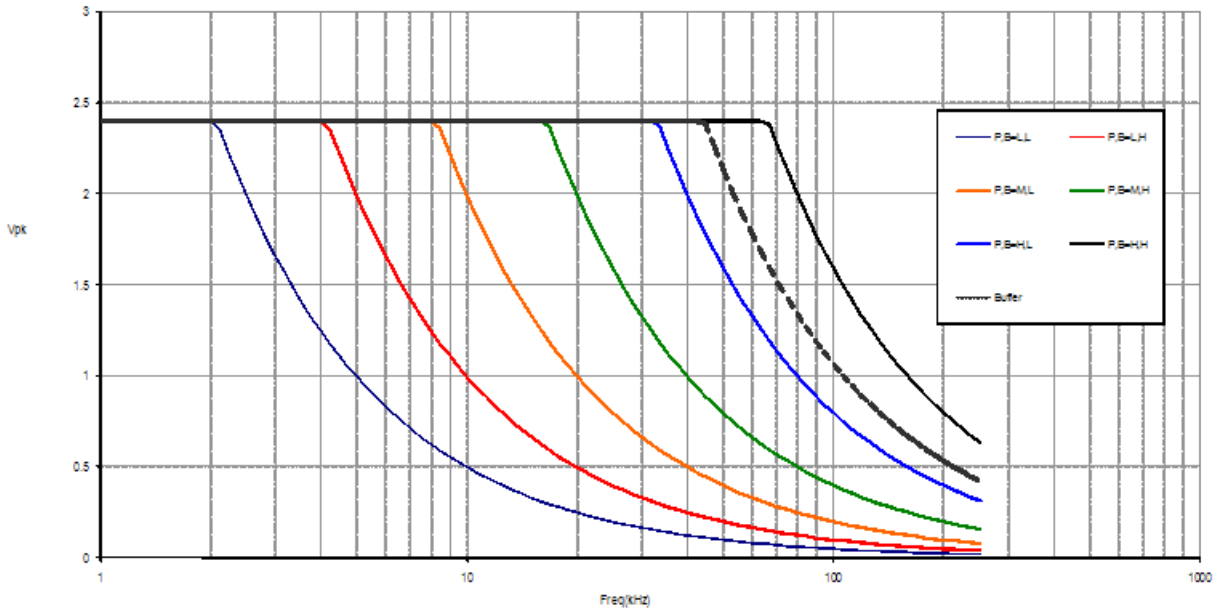
$$SR = 4 \times (2\pi V_{pk} F)$$

The maximum operating frequency for rail-rail output for V_{DD} = 5.0 V is given in the following table:

Power,Bias	kHz
P,B=H,H	64
P,B=H,L	32

Power,Bias	kHz
P,B=M,H	16
P,B=M,L	8
P,B=L,H	4
P,B=L,L	2

In this table, P = Power, B = Opamp Bias, L = Low, H = High and M = Medium. For lower signal levels, the allowed peak voltage is shown in the following graph:



The filter output is an internal signal. It must be buffered if driven off-chip. The buffer has a slew rate of 0.65 V/µsec. The operating frequency limits of the buffer are shown in the previous graph.

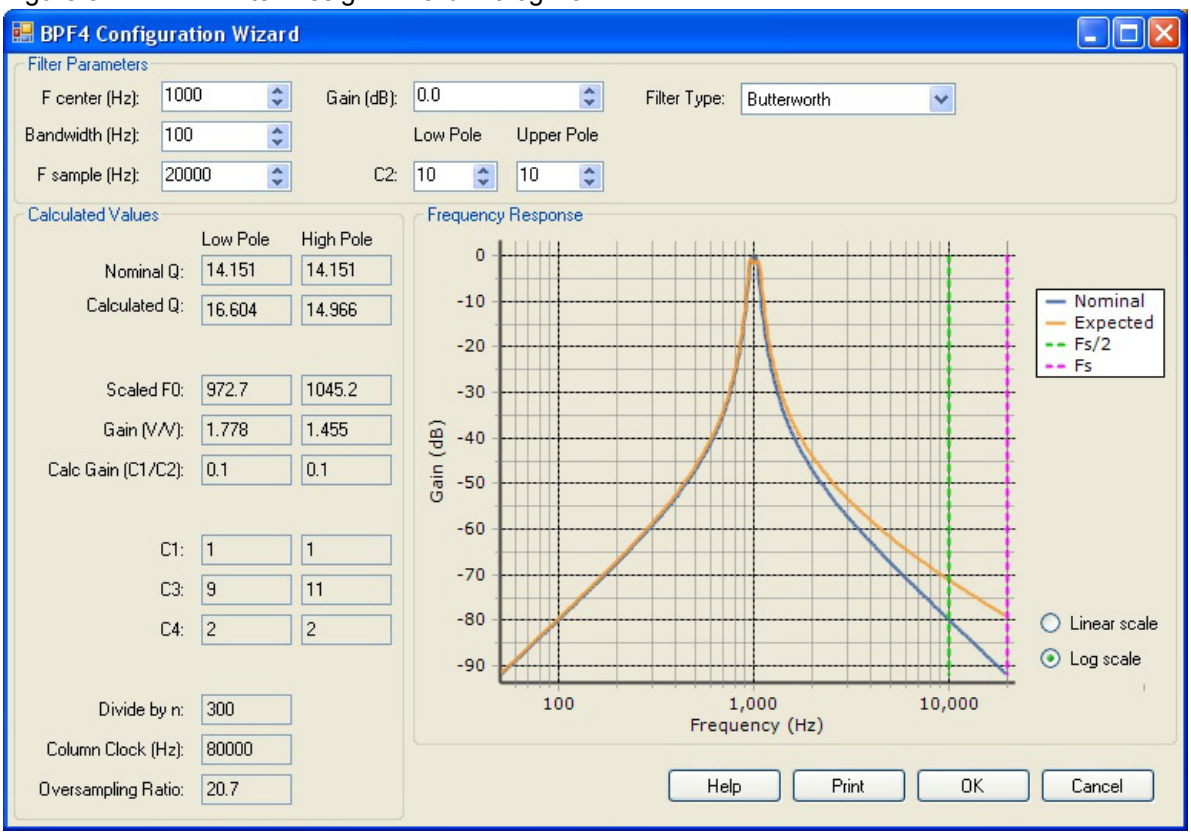
Filter Design

The design objective is to achieve the highest possible fc for the best waveform fidelity and minimum aliasing of out-of-band signals. Other system requirements may determine sample rates; capacitor values may be tailored to achieve the required sample rate.

The BPF4 provides three alternatives for determining the capacitor values. PSoC designer gives a filter design wizard to automate the procedure for four-pole filters. This same procedure is implemented in the spreadsheet, *BPF4 Design.xls*, which may be obtained from the “Documentation...” entry in PSoC Designer’s Help menu. Design constraints enforced by the wizard may be modified experimentally in the spreadsheets. For the ultimate in hands-on control over the design process, see the Appendix at the end of this user module datasheet for a numerical procedure that may be carried out manually. It also provides an example showing how the procedure works for a Butterworth filter with a 1 kHz corner frequency.

To use the PSoC Designer's built-in Filter Design Wizard, first place an BPF4 instance in the analog array. Right-click on the user module and choose "BPF4 Wizard..." from the pop up menu. The resulting dialog box, shown in Figure 5, describes a simple iterative procedure for designing the transfer function.

Figure 5. BPF4 Filter Design Wizard Dialog Box

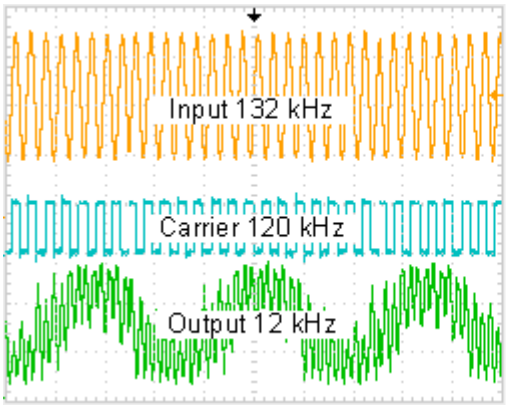


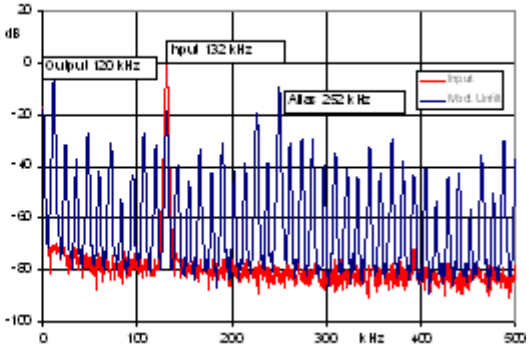
Scrolling down in the dialog box reveals the table of values used to plot the magnitude response. Values from this table may be cut and pasted into spreadsheets or other tools for further graphing and analysis.

Polarity and Modulator Functions

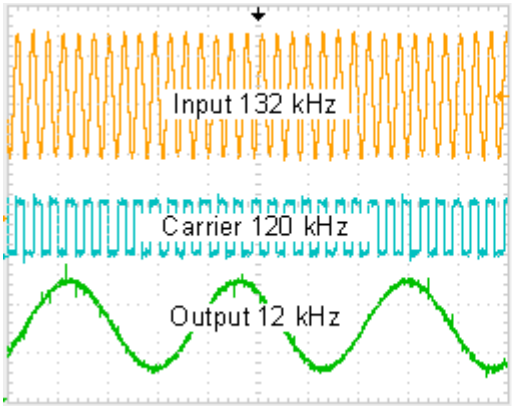
The BPF4's input switched capacitor block has a polarity control selection. The polarity can also be controlled by a signal external to the block to form a modulator. The modulator is used to shift frequencies up or down as the application requires. The modulation function works only in the first input block. The modulation function multiplies the input by +1 or -1 at the carrier rate. This generates signals at the sum of the carrier and input frequencies and at the difference of the carrier and input frequencies. The modulating square wave has harmonics at odd multiples of its fundamental. These harmonics fall off as 1/n so that aliases of the carrier are attenuated by the same factor. Modulation carriers should have 50% duty cycle to minimize even harmonic aliases.

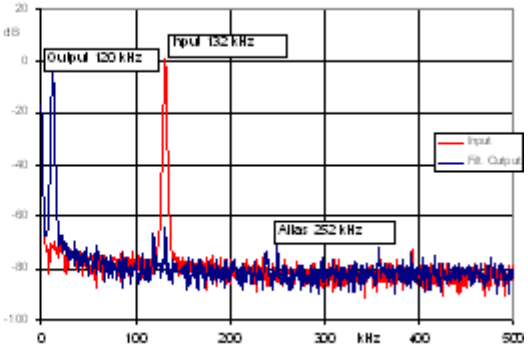
An example down-shift uses a 132 kHz signal, modulated by a 120 kHz square wave to generate a 12 kHz difference signal. An equal sized signal is generated at 252 kHz, but this is substantially filtered out by the band-pass filter. An example of a down-modulated signal without filtering is shown in the following figures:





The BPF4 configured at the difference frequency of 12 kHz with a 4 kHz bandwidth substantially eliminates harmonics and aliases, resulting in the following waveform. The modulation and half of the filtering are accomplished in the same block.





DC and AC Electrical Characteristics

Values given 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.

Table 1. 5.0 V BPF4 DC Electrical Characteristics

Parameter	Typical	Limit	Units	Conditions and Notes
DC Offset Voltage[1]	28	--	mV	Reference to Analog Ground[1]
Operating Current				
Low Power	290	--	µA	
Medium Power	1095	--	µA	
High Power	4200	--	µA	

Table 2. 5.0 V BPF4 AC Electrical Characteristics

Parameter	Typical	Limit	Units	Conditions and Notes
Mid-band Gain	.25	--	dB	Deviation from expected[2]
Maximum Clock Frequency[3]				
Low Power	.9	--	MHz	
Medium Power	4	--	MHz	
High Power	6	--	MHz	
Corner Frequency Error	.85	--	%	Deviation from Nominal[2]
Damping Ratio Error	1.05	--	%	
Noise[4]	615	--	nV/√Hz	

Values given are indicative of expected performance and based on initial characterization data. Unless otherwise specified in the following tables, limits guaranteed for TA = 25 °C, VDD = 3.3 V, Power HIGH, Opamp Bias LOW, output referenced to Analog Ground = VDD/2.

Table 3. 3.3 V BPF4 DC Electrical Characteristics

Parameter	Typical	Limit	Units	Conditions and Notes
DC Offset Voltage[1]	21	--	mV	Reference to Analog Ground[1]
Operating Current				
Low Power	270	--	µA	
Medium Power	1035	--	µA	
High Power	4045	--	µA	

Table 4. 3.3 V BPF4 AC Electrical Characteristics

Parameter	Typical	Limit	Units	Conditions and Notes
Mid-band Gain	.75	--	dB	Deviation from expected[2]
Maximum Clock Frequency[3]				
Low Power	.7	--	MHz	
Medium Power	1.3	--	MHz	
High Power	2.4	--	MHz	
Corner Frequency Error	1.55	--	%	Deviation from Nominal[2]
Damping Ratio Error	1.05	--	%	
Noise[4]	1150	--	nV/√Hz	

Electrical Characteristics Notes

1. Typical DC offset found using 1 kHz filter with Q's of 3, 5 and 15; C2=1 through 16; C3=3, 10 and 25. C1 and C4 found using filter design spreadsheet.
2. Deviation values determined from nominal filter: fcenter=1 kHz Butterworth, unity gain, C1=1, C2=3, C3=31, C4=1, fclock=20.3 kHz, Q=10.
3. Sample rate is one fourth of column clock frequency.
4. Noise found at 1 kHz using a 10 kHz filter.

Placement

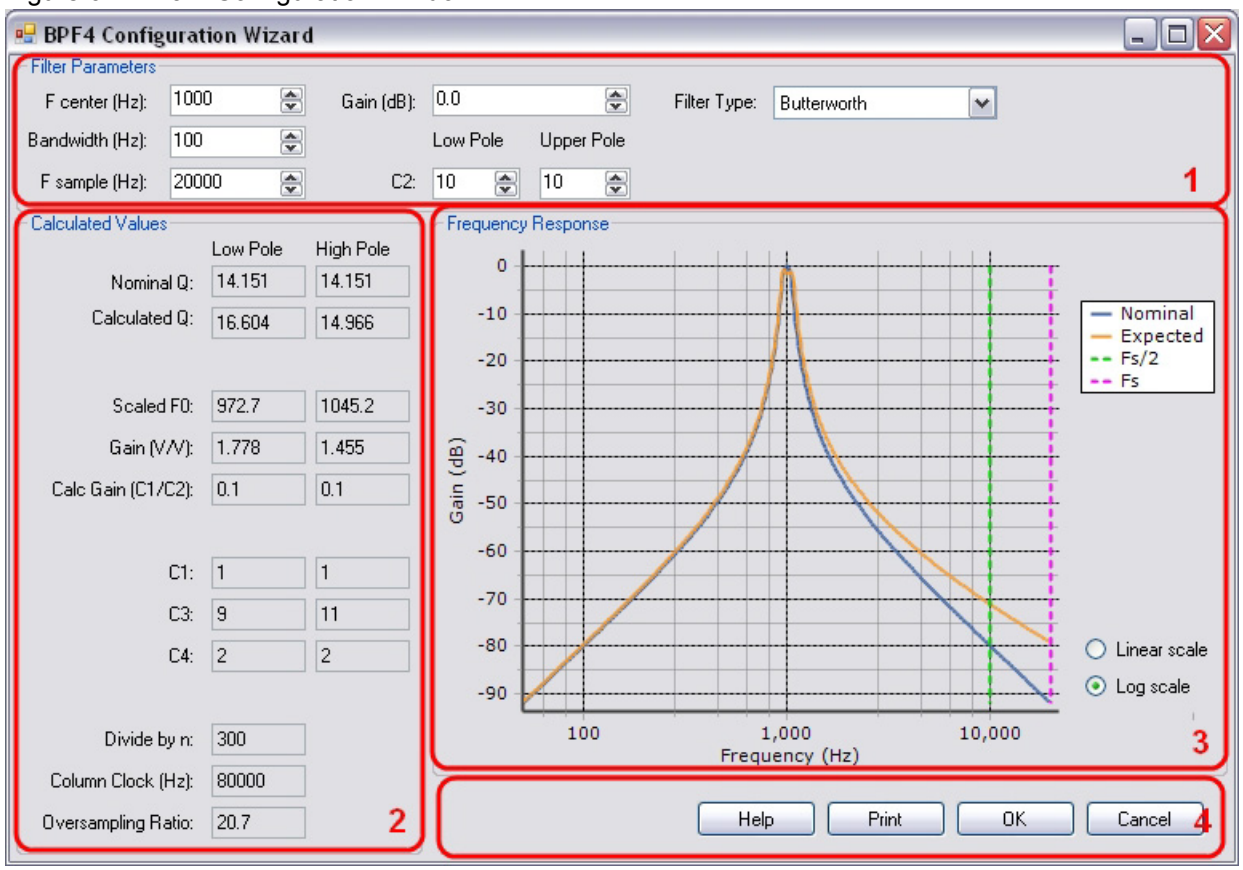
The Device Editor maps the logical FLIN and FLFB blocks onto a pair of adjacent switched capacitor PSoC blocks in the device's analog array. BPF4 UM uses two pairs of the FLIN and FLFB blocks named FLINL, FLFBL for first pair and FLINH, FLFBH for second. So BPF4 takes four SC blocks per one instance. There are several ways to construct the four-pole band-pass filter circuit out of the analog PSoC blocks. Each construction uses different capacitors and connections within the FLIN and FLFB 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 FLINH block.

Each time an instance of the BPF4 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 "Selection Options..." from the popup menu. Changing the topology after placement requires that the user module be placed in the analog array again.

Wizard

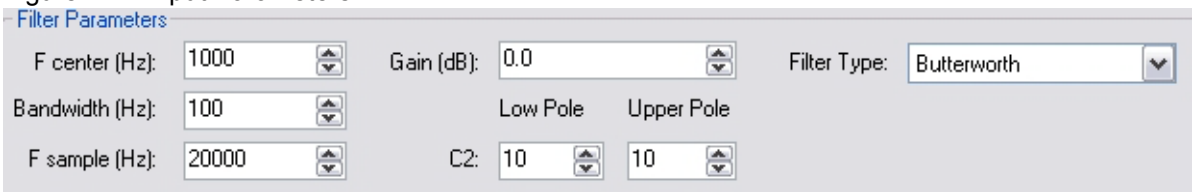
The filter configuration wizard window has the following sections: filter input parameters panel, filter output parameters panel, filter frequency response plot panel, and the wizard controls panel. Each parameters is described in this section.

Figure 6. Main Configuration Window



Field Input Parameters and Characteristics

Figure 7. Input Parameters



F Center

Enter desired center frequency for your filter.

Bandwidth

This is the desired bandwidth for your filter.

F Sample

This is the desired sample frequency for your filter.

Gain (dB)

Enter the desired gain for your filter in dB.

C2

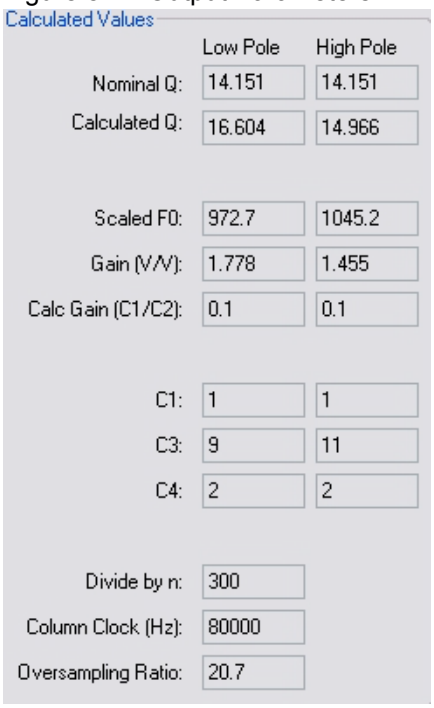
Enter the C2 values of your filter in range [1,31]. The value in the field marked Low Pole is transferred to the C2L user module parameter. The value in the field marked Upper Pole is transferred to the C2H user module parameter.

Filter type

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

Filter Output Parameters and Characteristics

Figure 8. Output Parameters



Nominal Q

This is the Q-factor required for the filter for lower and higher pairs of filter SC blocks respectively.

Calculated Q

This is the expected Q-factor of the designed filter for lower and higher pairs of SC blocks respectively.

Scaled F0

This is the expected roll-off frequency of the designed filter for lower and higher pairs of SC blocks respectively.

Gain(V/V)

This is the requested voltage gain of the filter for lower and higher pairs of SC blocks respectively.

Calculated Gain (C1/C2)

This is the expected voltage gain of the designed filter for lower and higher pairs of filter SC blocks respectively.

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

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 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

This is the calculated divider for the filter input clock. Use this divider number to configure PSoC clock dividers and the column clock where the filter is placed.

Column Clock (Hz)

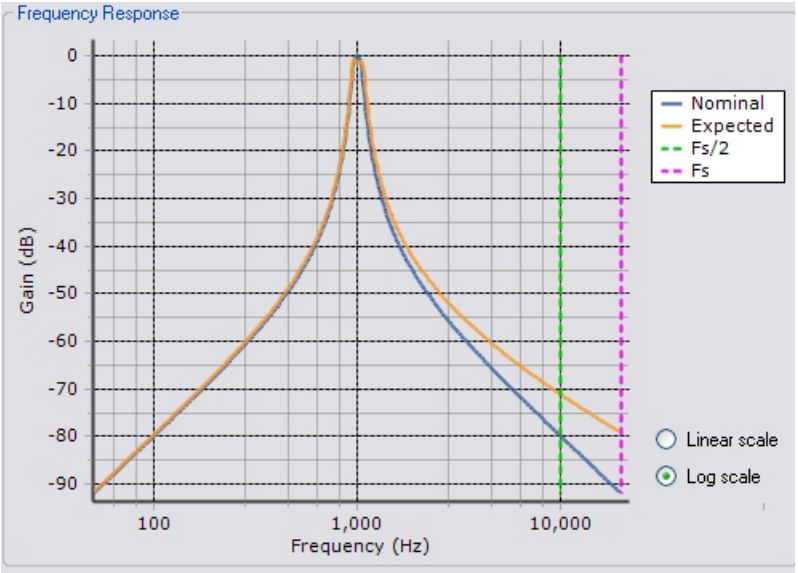
This is the calculated column clock for designed filter. Use this value to configure PSoC clock dividers and the column clock where the filter is placed.

Oversampling Ratio

The oversampling ratio of designed switched-capacitor filter. For lower and higher pairs of SC-blocks respectively.

Filter Response Plot

Figure 9. Filter Frequency Response Plot



Frequency response of designed filter

This is the area where the two frequency response plots are displayed – one for the nominal (desired) filter and another for the expected (can be implemented on PSoC).

Frequency scale selector

This parameter selects the frequency axis scale that is most representative for you.

Wizard Controls

Figure 10. Configuration Wizard Controls Panel



OK

Press this button to apply the parameters of the designed filter to the BPF4 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 the parameters and plot data.

Cancel

This button closes the Filter Configuration window without any modifications to the BPF4 User Module.

Parameters and Resources

To make a band-pass filter, place an instance of the BPF4 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 FLINH 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 BPF4 Block Diagram (Figure 1). 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 "BPF4 Wizard..." from the pop up menu. See the Filter Design section, for more information on design.

Modulator Clock

This property provides modulator source selection. Four sources are available for user selection:

Table 5. Modulator Clock source selection

Parameter	Note
None	Modulator Clock is OFF
GlobalOutOdd_0	Get Modulator Clock from GlobalOutEven_0
GlobalOutOdd_1	Get Modulator Clock from GlobalOutEven_1
Row_0_Broadcast	Get Modulator Clock from Row_0_Broadcast

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.

C2

This property is available from the GUI only. It provides C2 capacitor value setting. The property is divided onto two poles “Low Pole” and “Upper Pole”.

Center Frequency

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

Bandwidth

This property is available from the GUI only. It provides bandwidth frequency selection in the range from 10 Hz to 75 kHz.

Filter type

This property is available from the GUI only. It provides selection of classical all-pole filter configurations (Butterworth, Gaussian, Bessel, and Chebyshev).

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 BPF4 User Module, change power settings, and disable the user module.

BPF4_Start

Description:

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

C Prototype:

```
void  BPF4_Start(BYTE bPowerSetting)
```

Assembly:

```
mov   A, bPowerSetting
call  BPF4_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
BPF4_LOWPOWER	1
BPF4_MEDPOWER	2
BPF4_HIGHPOWER	3

Note For proper performance, filters with corner frequencies above 40 kHz should (1) use BPF4_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.

BPF4_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  BPF4_SetPower(BYTE bPowerSetting)
```

Assembly:

```
mov   A, bPowerSetting
call  BPF4_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.

BPF4_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  BPF4_SetC1L(BYTE bCapValue)
void  BPF4_SetC2L(BYTE bCapValue)
void  BPF4_SetC3L(BYTE bCapValue)
void  BPF4_SetC4L(BYTE bCapValue)
void  BPF4_SetC1H(BYTE bCapValue)
void  BPF4_SetC2H(BYTE bCapValue)
void  BPF4_SetC3H(BYTE bCapValue)
void  BPF4_SetC4H(BYTE bCapValue)
```

Assembly:

```
mov   A, bCapValue
call  BPF4_SetC1L                ; or, call  BPF4_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 BPF4 Schematic Drawing). Values outside this range are truncated to 31.

Return Values:

None

Side Effects:

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

BPF4_Stop

Description:

Powers the user module off.

C Prototype:

```
void  BPF4_Stop(void)
```

Assembly:

```
call  BPF4_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 band-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 BPF4 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[3] and the output is routed to P0[5].
//
//The following changes need to be made to the default settings in the Device Editor:
//
//   1. Choose the BPF4AS MUM configuration of the BPF4 user module.
//   2. Place it onto ASC10, ASD11, ASD20 and ASC21 blocks.
//   3. Rename User Module's instance name to BPF4.
//   4. Run BPF4 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, Filter Type and C2 parameters by default ("0.0", "Butterworth"
// and "10" respectively)
//      - Click the "OK" button
//   5. Set the BPF4's Input parameter to ACB00.
//   6. Set the BPF4's AnalogBus parameter to AnalogOutBus_1.
//   7. Leave the rest of UM parameters by default.
//   8. Set the AnalogColumn_Clock_0 and AnalogColumn_Clock_1 to VC2 (on the
//  interconnect view)
//   9. Set the AnalogOutBuf_1 to Port_0_5 (on the interconnect view).
//   10. Place the PGA UM onto ACB00 block.
//   11. Rename User Module's instance name to PGA.
//   12. Set PGA's Gain parameter to 0.125.
//   13. Set PGA's Input parameter to AnalogColumn_InputMUX_0.
//   14. Set PGA's Reference parameter to AGND.
//   15. Set PGA's AnalogBus parameter to Disable.
//   16. Set AnalogColumn_InputMux_0 to Port_0_3 (on the interconnect view).
//
```

```
// CONFIGURATION DETAILS:
//
//   1. The UM's instance names have to be shortened to BPF4 and PGA.
//   2. The Analog Column clock should be 250 kHz to get the 1kHz low Pass Filter with
//   Over Sampling Ration = 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
//  ------------------------------------------------------------------------------
//  BPF4       Name              BPF4                  UM's instance name
//             Input             ACB00
//             C1L               1                     Set by Wizard
//             C2L               10                    Set by Wizard
//             C3L               1                     Set by Wizard
//             C4L               7                     Set by Wizard
//             C1H               1                     Set by Wizard
//             C2H               10                    Set by Wizard
//             C3H               1                     Set by Wizard
//             C4H               7                     Set by Wizard
//             Modulator Clock   None
//             AnalogBus         AnalogOutBus_1
//
//  PGA        Name              PGA                   UM's instance name
//             Gain              0.125
//             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
    BPF4_Start(BPF4_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 BPF4 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[3] and the output is routed to P0[5].
;
;The following changes need to be made to the default settings in the Device Editor:
;
;   1. Choose the BPF4AS MUM configuration of the BPF4 user module.
;   2. Place it onto ASC10, ASD11, ASD20 and ASC21 blocks.
;   3. Rename User Module's instance name to BPF4.
;   4. Run BPF4 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, Filter Type and C2 parameters by default ("0.0", "Butterworth"
; and "10" respectively)
;      - Click the "OK" button
;   5. Set the BPF4's Input parameter to ACB00.
;   6. Set the BPF4's AnalogBus parameter to AnalogOutBus_1.
;   7. Leave the rest of UM parameters by default.
;   8. Set the AnalogColumn_Clock_0 and AnalogColumn_Clock_1 to VC2 (on the
; interconnect view)
;   9. Set the AnalogOutBuf_1 to Port_0_5 (on the interconnect view).
;   10. Place the PGA UM onto ACB00 block.
;   11. Rename User Module's instance name to PGA.
;   12. Set PGA's Gain parameter to 0.125.
;   13. Set PGA's Input parameter to AnalogColumn_InputMUX_0.
;   14. Set PGA's Reference parameter to AGND.
;   15. Set PGA's AnalogBus parameter to Disable.
;   16. Set AnalogColumn_InputMux_0 to Port_0_3 (on the interconnect view).
;
; CONFIGURATION DETAILS:
;
;   1. The UM's instance names have to be shortened to BPF4 and PGA.
;   2. The Analog Column clock should be 250 kHz to get the 1kHz low Pass Filter with
; Over Sampling Ration = 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
;  ------------------------------------------------------------------------------
;  BPF4       Name              BPF4                 UM's instance name
;             Input             ACB00
;             C1L               1                    Set by Wizard
;             C2L               10                   Set by Wizard
;             C3L               1                    Set by Wizard
;             C4L               7                    Set by Wizard
;             C1H               1                    Set by Wizard
;             C2H               10                   Set by Wizard
```

```
;              C3H               1                     Set by Wizard
;              C4H               7                     Set by Wizard
;              Modulator Clock   None                  Default value
;              AnalogBus         AnalogOutBus_1
;
;  PGA         Name              PGA                   UM's instance name
;              Gain              0.125
;              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, BPF4_HIGHPOWER
    lcall BPF4_Start

    ; Insert your main assembly code here.

.terminate:
    jmp .terminate
```

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 BPF4_SetC1L API function may be used to implement a programmable-gain control.

Configuration Registers

Both the Horizontal and Vertical topologies have similar bit fields. Their description follows the tables.

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	C4L				
CR3	0	0	1	0	Feedback		Power	

Table 7. Block FLFBL: Register CR

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

Table 8. Block FLINH: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CAH	0	Polarity	C1H				
CR1	FLFBL			C2H				
CR2	AnalogBus	CompBus	0	C4L				
CR3	0	0	1	0	Feedback		Power	

Table 9. Block FLFBH: Register CR

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

Vertical A-Input Topology

Table 10. 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 11. Block FLFBL: Register CR

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

Table 12. Block FLINH: Register CR

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

Table 13. Block FLFBH: Register CR

Bit	7	6	5	4	3	2	1	0
CR0	CBH	0	0	C3H				
CR1	FLINH			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:

CAL, CAH, CBL and CBH

Set the FLINL, FLINH, FLFBL and FLFBH feedback capacitors, respectively, to either 16 or 32 units. (See Figure 1)

C1L, C2L, C3L, C4L, CPPL, C1H, C2H, C3H and C4H

Set the capacitors illustrated in the BPF4 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 BPF4 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 BPF4 User Module in the Device Editor.

FLINL

Is the connection from the FLINL output to the FLFBL input, automatically determined by placement of the BPF4 User Module in the Device Editor.

FLFBL

Is the connection from the FLFBL output to the FBINH input, automatically determined by placement of the BPF4 User Module in the Device Editor.

FLINH

Is the connection from the FLINH output to the FLFBH input, automatically determined by placement of the BPF4 User Module in the Device Editor.

Polarity

Controls whether the output of the filter is inverted or not. This property is not used in the current user module.

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 BPF4_Start and can be modified by calling the functions BPF4_SetPower and BPF4_Stop.

Appendix: Numerical Design Procedure

The following manual design procedure for one pole-pair filters is automated in the 2-pole Band-pass Filter Design Wizard built into the Device Editor. To start the Wizard, right click on a a BPF2 filter that has been placed in the analog array and choose “Filter Design Wizard…” from the pop-up menu. When the transfer function is satisfactory, clicking the Wizard’s “OK” button transfers the calculated C1 through C4, CA, and CB values into the Device Editor parameters. In addition, you can use the Microsoft Excel spreadsheet, *BPF2Design.xls*, located in the PSoC Designer documentation directory, for computation and graphical analysis of the transfer function.

Four-Pole Design Procedure

1. Determine filter requirements for center frequency, Q and mid-band gain. If upper and lower -3 dB points, fu and fl are known, calculate center frequency and Q from $f_c = \sqrt{f_u f_l}$ and $Q = \frac{f_c}{f_u - f_l}$
2. Set CA and CB equal to 32.
3. Set initial value of C2 equal to the Q of the pole pair.
4. Set initial value of C3 equal to the Q of the pole pair.
5. Calculate value for C4 from rearranged Equation 4, solved for the quadratic.

Equation 10

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

6. C4 must be real and positive. Round to the nearest integer.
7. If C4 is determined to be negative or imaginary, adjust the value of C3, then calculate a new value for C4. Iterate as necessary.
8. Calculate the sample frequency from rearranged Equation 5.
9. Evaluate the oversample ratio. $$OSR = \frac{f_s}{f_c}$$
10. If OSR is less than 5.0, adjust C2 and repeat steps 3 through 9.
11. Calculate value for C1 from Equation 6. Round to nearest integer.
12. Calculate analog column clock, equal to four times fs. Pick an analog column clock resource, fsysclk, from the selections available in PSoC Designer. See the Sample Clock parameter for additional details on clock selection. Note that the BPF2 analog switched capacitor PSoC blocks are in separate columns. Both blocks, thus both columns, must have the same clock.
13. Calculate the divider by rounding to the nearest integer. $$n = int\left(\frac{f_{sysclk}}{4f_{clk}} + 0.5\right)$$
14. Divide the selected system clock resource by 4n, to get the actual clock frequency.
15. Calculate the filter center frequency based on the actual sample clock.
16. Calculate Q and Gain, G.
17. Evaluate filter design performance. Design values for fc, Q and G can be realized typically within 3% of design requirements in almost all cases. If the error is larger than this, select a new value for C2, iterate and optimize.

Version History

Version	Originator	Description
2.0	DHA	1. Excluded Comparator Buses from the Modulator Clock Value List. 2. Replaced GlobalOutOdd_0" and "GlobalOutOdd_1" values of the "Modulator Clock" Property with "GlobalOutEven_1" and "GlobalOutEven_0". 3. Corrected "Nominal Q" calculations in wizard.
2.10	DHA	1. Added help file to wizard. 2. Updated images in this user module datasheet.
2.10.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 © 2009-2013 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.