FFTea

User Manual (Rev. 1.0)

Table of Contents
(Click on title for hyperlink)

Table of Contents

(Click on title for hyperlink)

FFTea is a product of:

CURASI SOFTWARE LAB
PO BOX 580545
ELK GROVE, CA 95758

cswlab.com

1.0 Application Overview

Digital filter design is fundamental in the design of Field Programmable Gate Array (FPGA) logic that utilizes digital signal processing techniques. Regardless of whether the filter is fixed-rate or multi-rate, there exists a need to easily design a linear phase filter and simulate its frequency response prior to implementation in the FPGA. FFTea was developed for ease of use for the experienced Engineer as well as the novice.

The key features of FFTea are:

FIR Filter Designer

- Linear Phase FIR Filters (non-recursive structure)
- Symmetric Filter Coefficients (positive or negative symmetry)
- Constant Group Delay (zero phase distortion)
- Low Pass, High Pass or Bandpass filters
- Filter Length: 25 to 255 taps (in odd increments)
- Fixed Precision: 20-bit to 8-bit (in integer decrements)
- Windowing functions include Kaiser and Chebyshev amongst others
- Magnitude and Unwrapped Phase Response
- Supports Normalized Units

Export of FIR Coefficients

- Generate a CSV file of Coefficients for any Fixed Precision Filter
- Filter Coefficients Scaled for Maximum Dynamic Range
- Ready for import by FPGA design tools

Filter Inspector

- A Tabular View of all previously designed FIR Filters
- All Filter Design Parameters and Design Results are Clearly Organized

Signal Designer

- Generate Waveforms at a specified Sampling Frequency
- Simply Select a Function from a Scroll and input its parameters
- Optional Additive Gaussian Noise for all functions
- Time and Frequency Domain Graphs

Signal Inspector

- A Tabular View of all previously generated Signals
- All Signal Parameters are Clearly Organized

Sum & Filter Model

- Test a Filter Design easily by specifying the Filter and its Input Signals
- Input can either be a Sum or Difference of two Signals, or a single Signal
- Time & Frequency Graphs for both the Input and the Filter's Response

Multirate Model

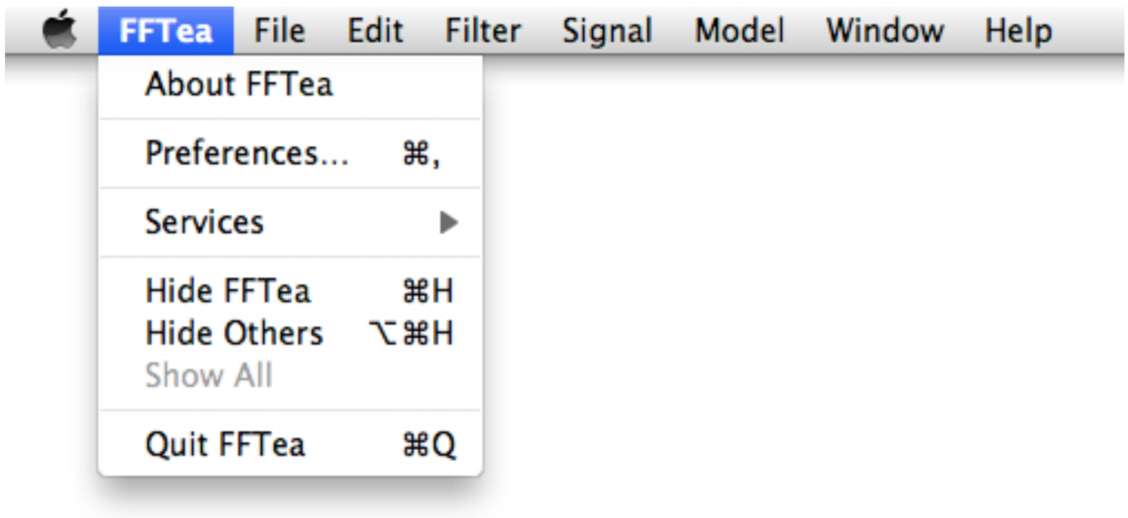
- Easily Test a Multirate Filter Design
- Specify both the Interpolate and Decimate Values: 1 to 10
- Optional auto-generated Filter allows Comparison to a Reference Design
- Input can either be a Sum or Difference of two Signals, or a single Signal
- Time & Frequency Graphs for both the Input and the Filter's Response

2.0 System Requirements

The System requirements for FFTea are as follows:

- Mac OS 10.6 (Snow Leopard) or Mac OS 10.5 (Leopard)
- An Intel Processor or a PowerPC (G4 or G5)
- Minimum Display Resolution: 1600 by 1000 pixels

3.0 FFTea Menu



3.1 About FFTea

Figure 3.1 illustrates the “About FFTea” Panel. Note that it contains the version number of the application.

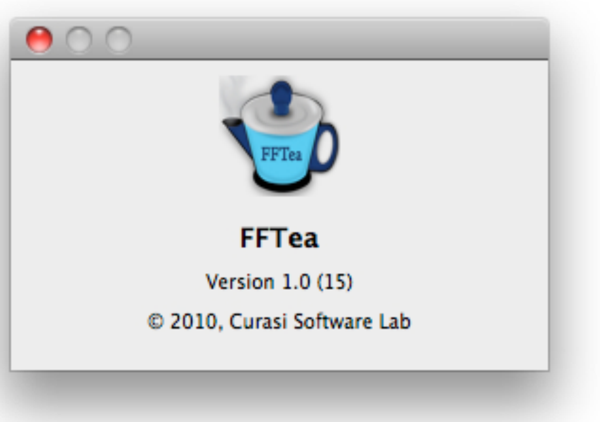


Figure 3.1 - About FFTea Panel

3.2 Preferences

FFTea's Preference Panel is shown in figure 3.2. A set of six checkboxes allow the user to specify which of FFTea's windows are open at the start of the application. The default configuration consisting of the Filter Designer, Filter Inspector and Signal Inspector is illustrated below. Both the Filter and Signal Inspectors are always useful to have open.

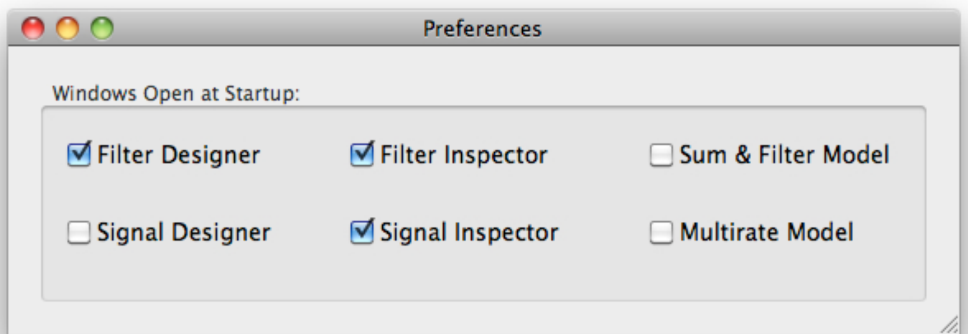
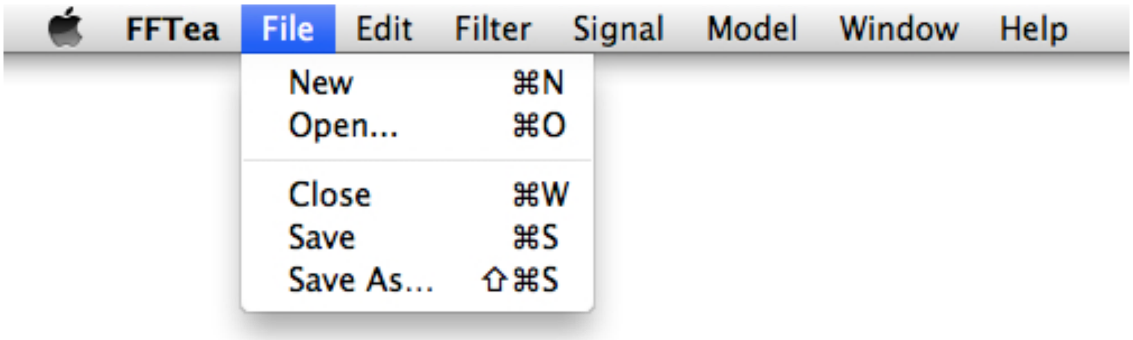


Figure 3.2 - Preferences Panel

4.0 File Menu



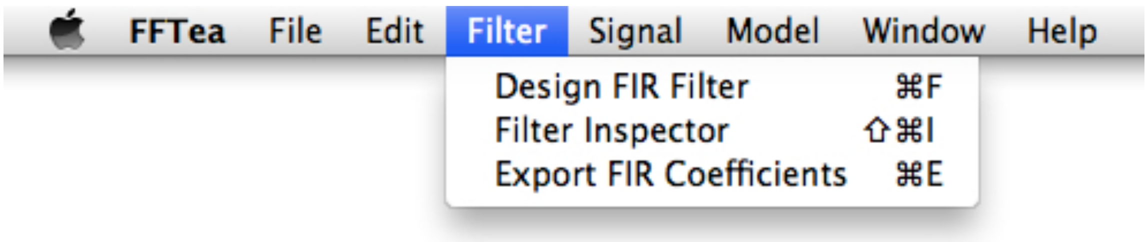
The File Menu's operations are applicable to the Filter, Signal, and Model windows as shown in table 4.0. If multiple windows are open, the key (front-most) window responds to the user's commands.

Section	Window Title	Key-Equivalent
5.1	Design FIR Filter	⌘F
6.1	Generate Signal	⌘G
7.1	Sum & Filter Model	⌘Y
7.2	Multirate Model	⌘U

Table 4.0 - Windows Supporting File Operations

User generated filter, signal, and model files each have a unique file structure (and file type) and are stored as part of FFTea's application support files. Consequently, the user needn't be concerned about misplacing either a filter, signal, or model file.

5.0 Filter Menu



The Filter menu's items allow the user to generate a FIR filter, export its coefficients, and view the design parameters of each previously designed filter.

5.1 Design FIR Filter

An example of the FIR Filter Designer window is illustrated in figure 5.1.1. This figure shows the magnitude response of a lowpass filter with a sampling rate of 300 MHz and a cutoff frequency of 25 MHz. Figure 5.1.2 illustrates the unwrapped phase response of the identical filter. Note that the filter's phase response is linear from 0 to -75 radians corresponding to the passband of the preceding figure.

The filename of the filter "lp300fc25" is indicated in the titlebar of both figures.

5.1.1 Filter Length

The length or number of taps of the filter is selected by the stepper control as shown in figure 5.1.1. The filter length may be set in odd increments from 25 to 255 taps. The filter's coefficients will always be symmetric regardless of the number of taps. For a symmetric filter h[n] with length N, positive symmetry is defined as:

$$h[n] = h[N - n - 1]$$

Similarly, for a symmetric filter with negative symmetry:

$$h[n] = -h[N - n - 1]$$

Since N is always odd the filter's center tap will always be unique.

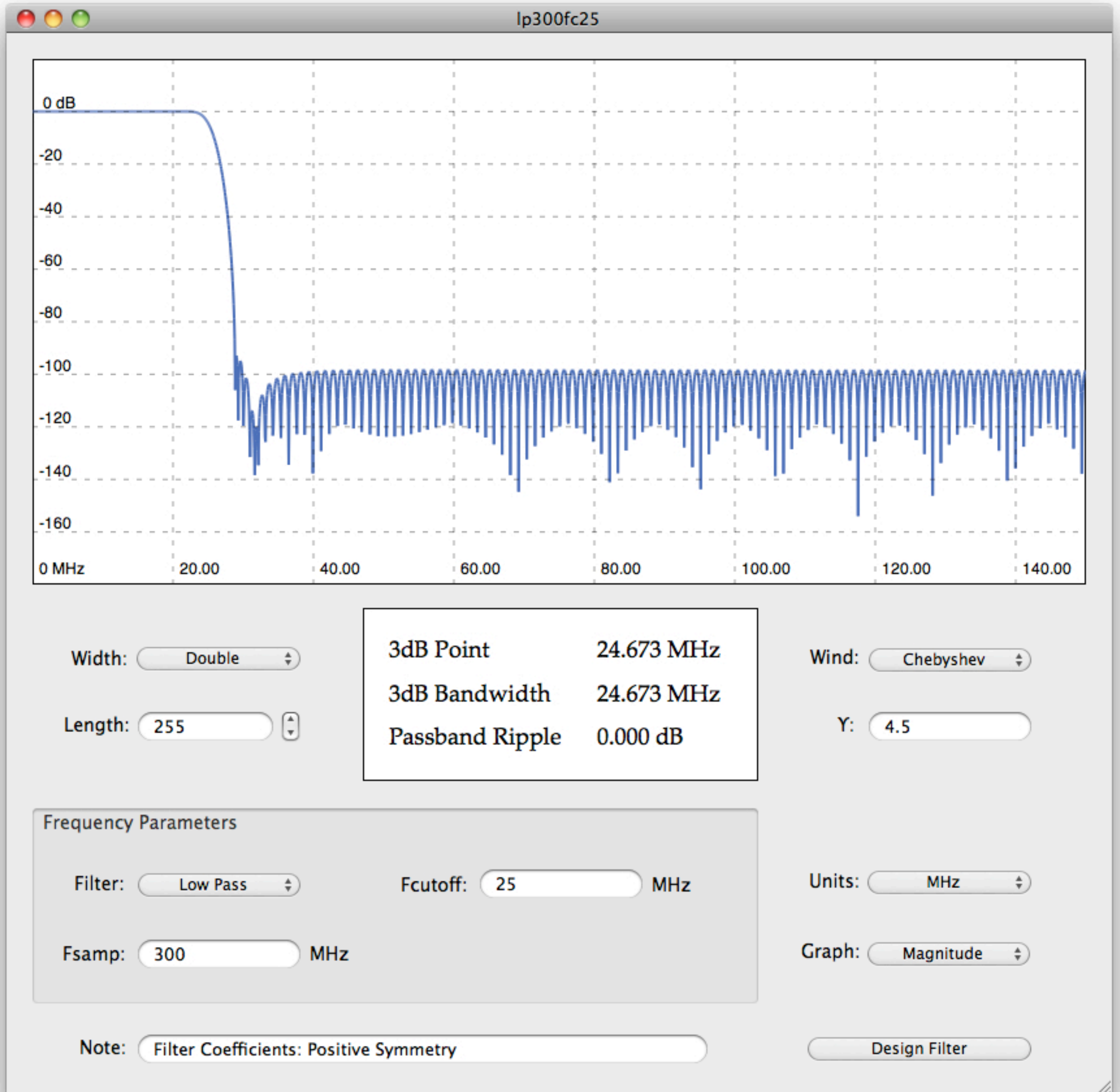


Figure 5.1.1 - FIR Filter Designer Window with Magnitude of a LPF

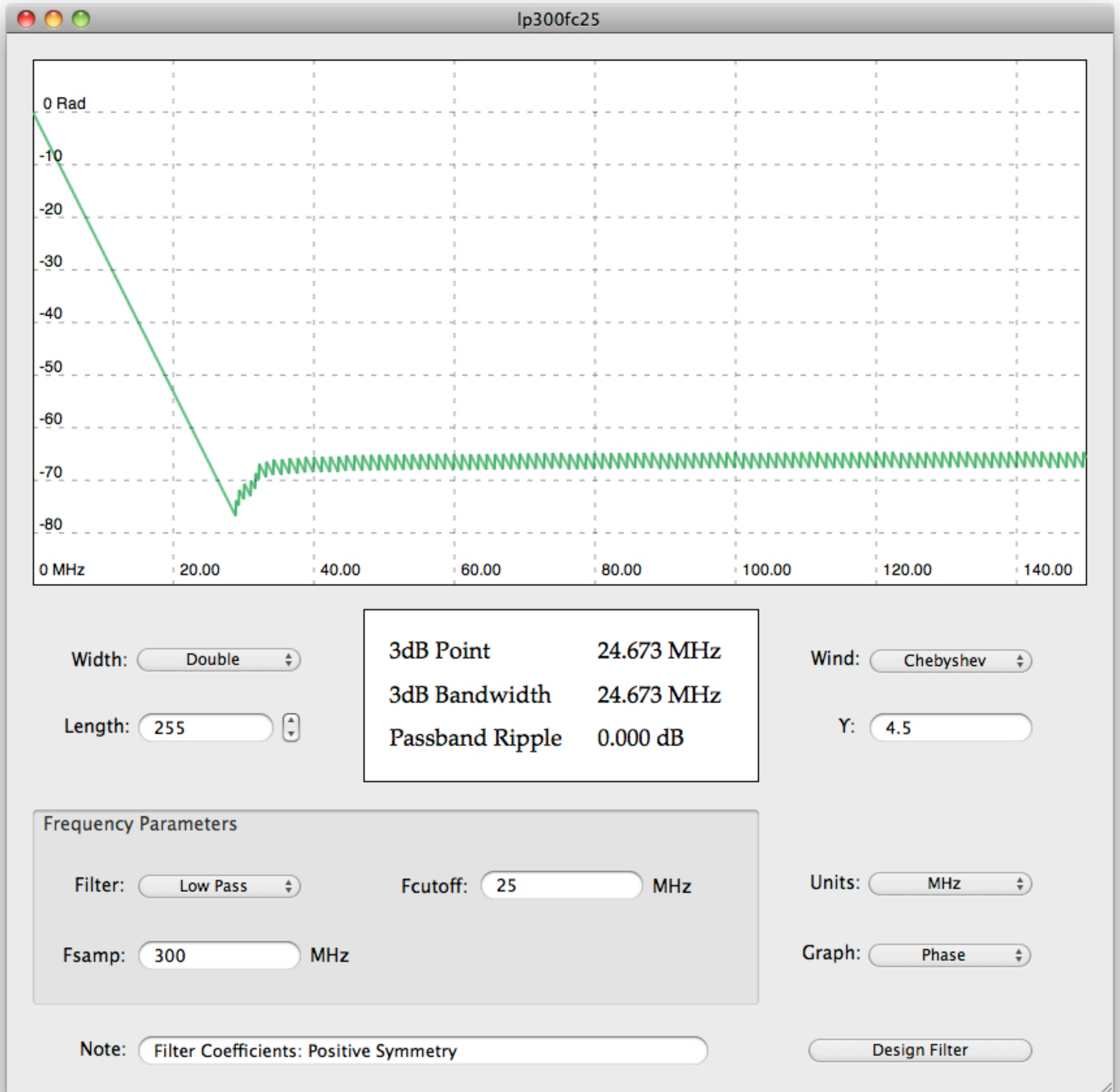


Figure 5.1.2 - FIR Filter Designer Window with Phase of a LPF

5.1.2 Filter Width

Referring to the above figure, note that the filter's width is set via a pop-up menu and set to "Double" for double precision floating point. In addition to floating point, fixed point widths from 20-bit to 8-bit (in integer decrements) may be selected.

When designing a filter, it is best to first use double precision to determine the optimal response of a given filter, since it does not limit the dynamic range as in the case of fixed-precision widths. After a filter design is established, then a fixed-point width is required to generate integer values for filter coefficients. The coefficients may be export for a FPGA design using the export utility in section 5.3.

5.1.3 Frequency Parameters

The Frequency Parameters consists of a pop-up for the selection of the filter type in addition to a group of up to three input fields as shown in figure 5.1.3. The selectable types consist of a low pass, high pass and band pass filter.

The magnitude and phase response of a band pass filter design are illustrated in figures 5.1.3 and 5.1.4 respectively. Band pass filters require a center frequency "Fcenter" in addition to the cutoff frequency "Fcutoff" required by all filters.

Regarding the band pass filter example, the frequency for both the magnitude and phase graphs span from 0 to a maximum of 25 MHz. Since the Nyquist Frequency for a signal at 25 MHz is twice the rate, the sampling frequency "Fsamp" is 50MHz. When choosing a sampling frequency, generally it is best to over-sample at a rate of at least 10x the maximum frequency component of a signal.

5.1.4 Frequency Units

The frequency units are selectable via a pop-up and may either be set to MHz, KHz, or Normal. The selected units apply to all frequency parameters.

When the frequency units are set to Normal, the frequency for both the magnitude and phase graphs span from 0 to unity. Since normalized units are relative rather than absolute, the Sampling Frequency input field is not relevant and consequently hidden.

Consider the following signal with an analog frequency f_a and a sampling rate of f_s:

$$x[n] = \cos(2 \pi f_a n T) \quad \text{where } T = 1/f_s$$

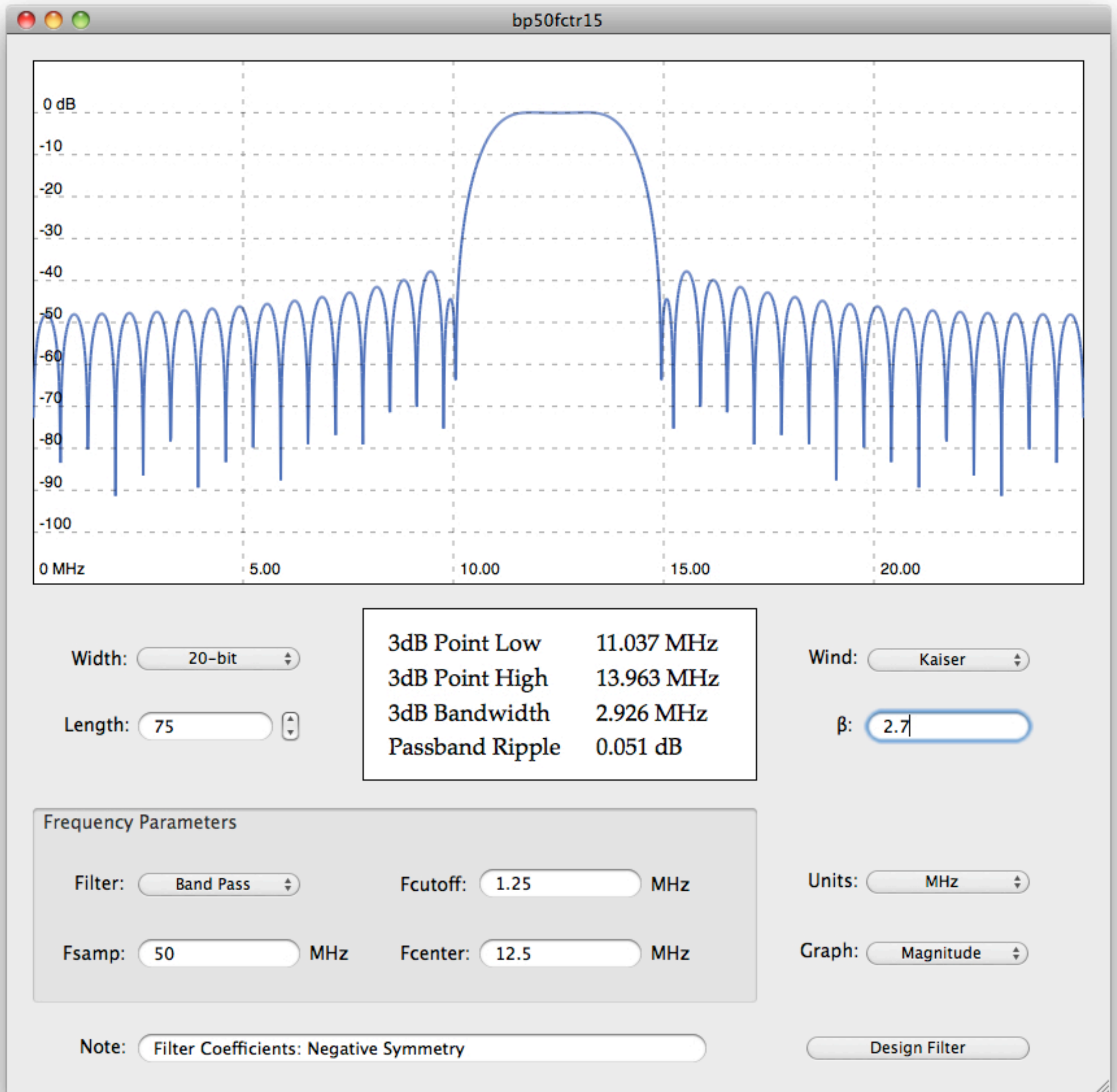


Figure 5.1.3 - FIR Filter Designer Window with Magnitude of a BPF

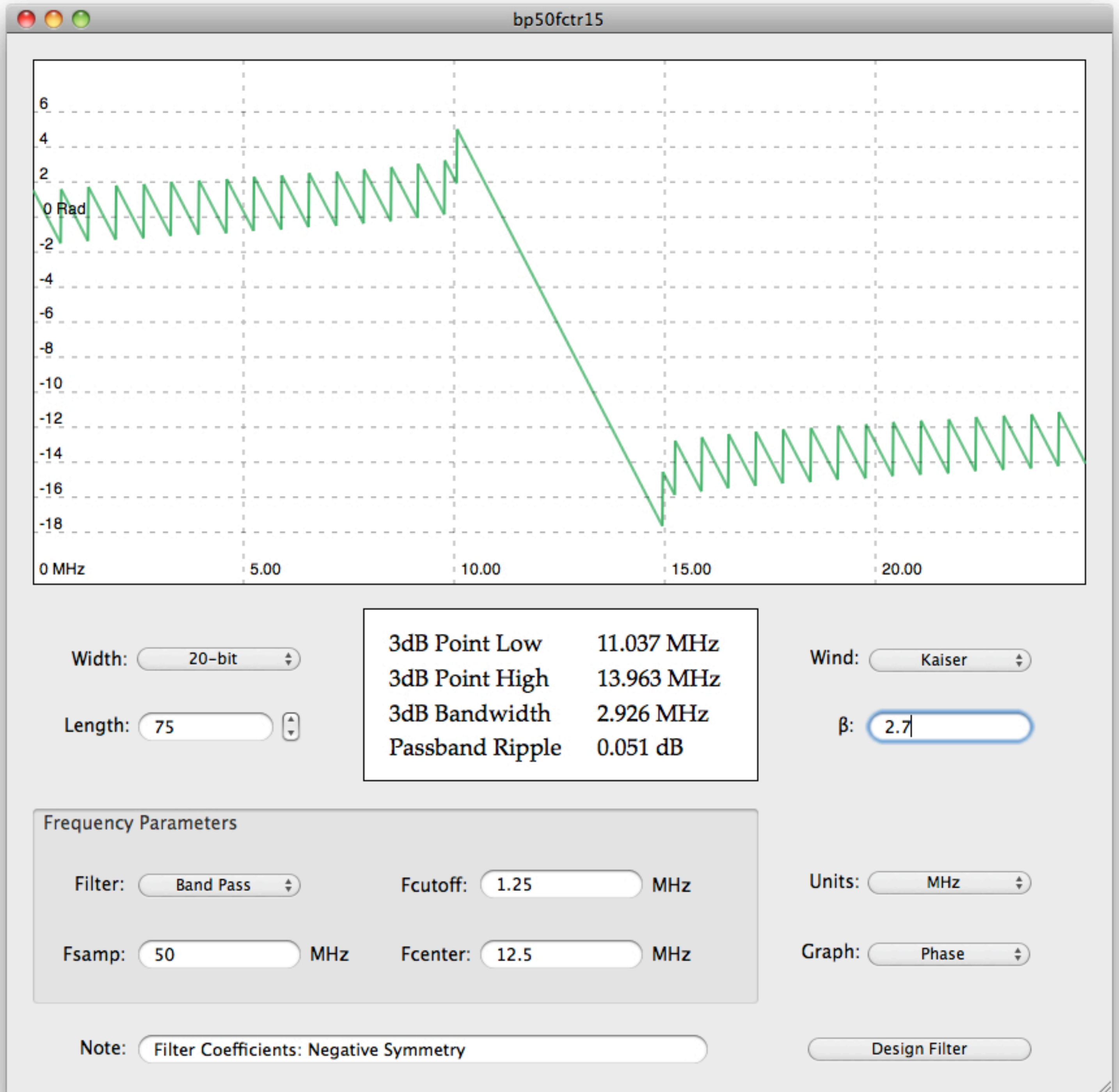


Figure 5.1.4 - FIR Filter Designer Window with Phase of a BPF

If the sampling frequency is set to the Nyquist rate:

$$f_s = 2\ f_a$$

The discrete signal x[n] simplifies to:

$$x[n] = \cos(\pi\ n)$$

As evident from the equation above, a filter designed with normalized units is independent of the sampling frequency. If the designer chooses to design filters with normalized units, the filter design may then be applied to any system with a given sampling frequency.

An example of a high pass filter with normalized units is shown in figure 5.1.5. Note that the cutoff frequency is 0.5, and recall that unity represents half the sampling frequency. Therefore, if the filter is utilized in a system with a sampling frequency of 100 MHz, the frequency of the magnitude graph would span from 0 to a maximum of 50 MHz. Since the filter is high pass, the passband of the filter would be from 25 to 50 MHz.

5.1.5 Windowing Functions

Windowing functions are weighting functions that can be used to reduce the sidelobes and ripple of a filter design. The choice of a windowing function depends upon the Engineer's design requirements. Table 5.1.5 lists the available window functions of the FIR Filter Designer that may be selected via the "Window" pop-up menu. In addition, the table lists the attenuation of the first sidelobe, the extent of the transition region, and the passband ripple for each windowing function. The measured values were for a low pass filter design with a length of 125 taps. Note that a steeper transition region can always be obtained by increasing the number of filter taps, at the cost of increased complexity.

As evident from table 5.1.5, the rectangular window function offers the designer the sharpest transition region at the penalty of increased ripple in the passband. In general, windowing functions with greatly reduced passband ripple have a wider transition region.

Two widely popular functions are the Hamming window and the Hann window. Both functions are raised cosines; however, the Hamming window has a DC offset and consequently is non-zero at its extremities. Both offer a good compromise with reduced passband ripple and a reasonable transition region.

The Nutall, Blackman-Harris, and Blackman-Nutall are high dynamic range windowing functions with negligible passband ripple. As a result, their transition regions are among the highest listed.

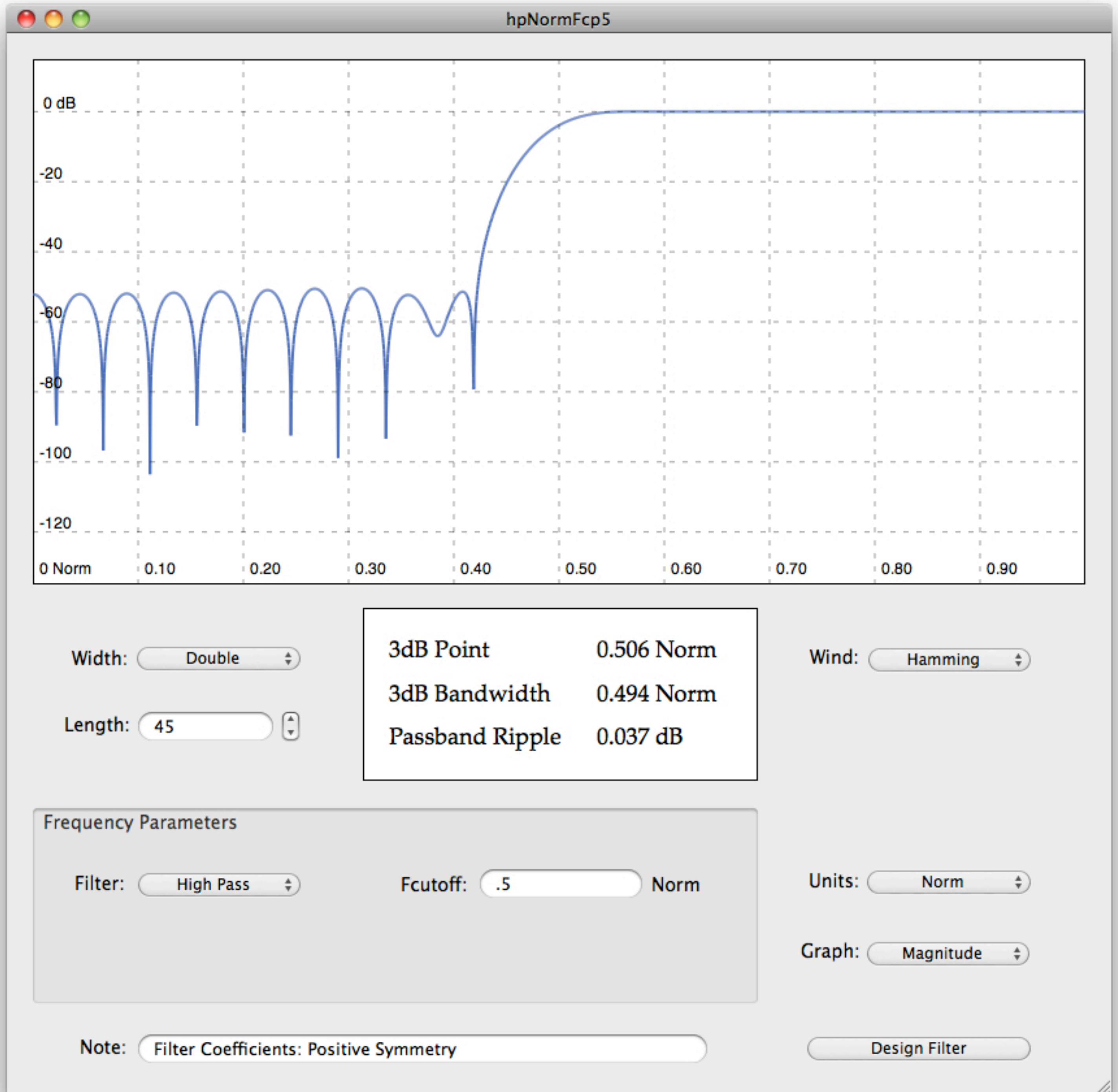


Figure 5.1.5 - FIR Filter Designer Window with Magnitude of a HPF

Window Function of Low Pass Filter	First Sidelobe		Passband Ripple (dB)
	Attenuation (dB)	**Transition Region** (radians/sample)	
Rectangle	-16.8	0.009 π	1.117
Hann	-40.0	0.026 π	0.089
Hamming	-48.8	0.027 π	0.036
Blackman	-71.9	0.043 π	0.002
Nutall	-100	0.060 π	0.000
Blackman-Harris	-106	0.060 π	0.000
Blackman-Nutall	-110	0.060 π	0.000
Kaiser (β=1)	-19.1	0.010 π	0.869
Kaiser (β=2)	-27.3	0.015 π	0.344
Kaiser (β=3)	-43.2	0.026 π	0.038
Kaiser (β=4)	-63.6	0.048 π	0.005
Kaiser (β=5)	-92.0	0.064 π	0.000
Chebyshev (γ=2)	-46.7	0.024 π	0.045
Chebyshev (γ=2.5)	-61.4	0.030 π	0.020
Chebyshev (γ=3)	-69.6	0.036 π	0.006
Chebyshev (γ=3.5)	-74.9	0.041 π	0.002
Chebyshev (γ=4)	-82.1	0.046 π	0.000
Chebyshev (γ=4.5)	-91.9	0.052 π	0.000

Table 5.1.5 - A Low Pass Filter Comparing Various Window Functions

The Blackman windowing function offers increased sideband attenuation over the Hamming and Hann windows, and a transition region that is substantially stepper than the high dynamic range windowing functions.

The Kaiser and Chebyshev windowing functions are categorically different from the aforementioned since they both offer a means of compromising between the steepness of the transition region and the sidelobe attenuation level. Figure 5.1.3 illustrates the magnitude of a bandpass filter using a Kaiser windowing function with 2.7 assigned to its β parameter. Note that the Kaiser window's sidelobes decrease the further away they are from the center frequency. Increasing the value of β decreases the sidelobe attenuation level and broadens the width of the main lobe. As the width of the main lobe increases, the transition region also increases. A comparison of the effect of varying β is shown in table 5.1.5.

The magnitude of a low pass filter utilizing a Chebyshev windowing function is illustrated in figure 5.1.1. The Chebyshev window's sidelobes have a stopband attenuation that is predominantly uniform and is a function of its γ parameter. Increasing the value of γ decreases the sidelobe attenuation level which undoubtably increases its transition region. A comparison of the effect of varying γ is shown in table 5.1.5.

5.1.6 Design Results View

The Design Results View for a high pass filter is shown at the center of figure 5.1.5. For both low pass and high pass filters, the view consists of the 3dB point, 3dB bandwidth and the passband ripple. The voltage at the 3dB point is 0.707x the passband voltage, and its power is one-half the passband power. The passband ripple is the maximum amount of ripple in the filter's passband and is measured in decibels.

The 3dB bandwidth is simply the passband bandwidth and is measured from the 3dB point to unity for the high pass filter shown in figure 5.1.5. For the case of a low pass filter, the 3dB bandwidth is measured from zero to its 3dB point and therefore both are equal (as shown in figure 5.1.1). A bandpass filter has both high and low 3dB points, and its bandwidth is the passband measured between these points as illustrated in figure 5.1.3.

5.1.7 Magnitude & Phase Graphs

Either a magnitude or unwrapped phase graph may be selected via the "Graph" pop-up menu. The y-axis of the magnitude graph has a maximum value of zero decibels, and its grid decrements in multiples of ten decibels. If the "Units" pop-up menu is set to normalized units, the x-axis spans from zero to unity. Refer to section 5.1.4 for a thorough description of using normalized units for frequency. If the frequency units are not normalized, the x-axis spans from a minimum of zero to a maximum of one-half the sampling frequency.

The unwrapped phase graph shows the true phase response of the filter in radians and is always linear over the filter's passband. Figures 5.1.2 and 5.1.4 illustrate the phase response of a low pass and bandpass filter respectively. Note that the x-axis of the phase graph is identical to that of the magnitude graph.

As a result of the phase linearity in the passband, the group delay is always constant which results in a filter design with zero phase distortion. Phase linearity is only possible with symmetrical filter coefficients. Filter coefficients may either have positive or negative symmetry as previously discussed in section 5.1.1.

Consider the digital frequency θ of a sampled waveform with analog frequency f_a and sampling rate f_s:

$$\theta = 2 \pi f_a / f_s$$

the phase ϕ of a filter with positive symmetric coefficients can be written as:

$$\phi(\theta) = - a \theta$$

likewise, the phase of a filter with negative symmetric coefficients:

$$\phi(\theta) = b - a \theta$$

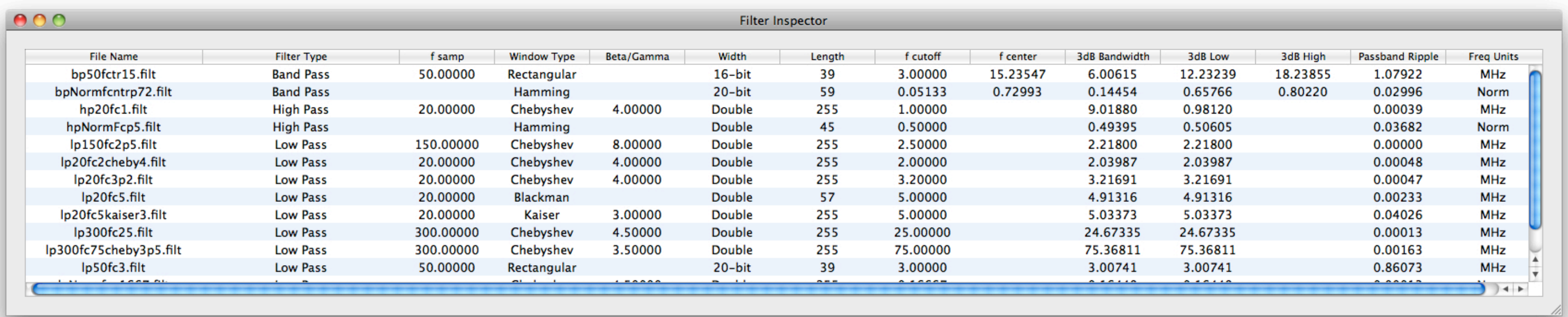
therefore the Group Delay of either filter is equivalent to:

$$\text{Group Delay} = - \frac{d \phi(\theta)}{d \theta} = a$$

A constant group delay indicates that signal components at different frequencies receive the identical time delay in the filter.

5.2 Filter Inspector

The Filter Inspector window summarizes all filter parameters and design results in a tabular view as shown in figure 5.2. By default, the Filter Inspector is positioned at the lower right corner of the monitor upon the start of the FFTea application. The Inspector serves as a reference for both the Fixed-Rate Filter Model (section 7.1) and the Multirate Filter Model (section 7.2).


Filter Inspector

File Name	Filter Type	f samp	Window Type	Beta/Gamma	Width	Length	f cutoff	f center	3dB Bandwidth	3dB Low	3dB High	Passband Ripple	Freq Units
bp50fctr15.filt	Band Pass	50.00000	Rectangular		16-bit	39	3.00000	15.23547	6.00615	12.23239	18.23855	1.07922	MHz
bpNormfcntrp72.filt	Band Pass		Hamming		20-bit	59	0.05133	0.72993	0.14454	0.65766	0.80220	0.02996	Norm
hp20fc1.filt	High Pass	20.00000	Chebyshev	4.00000	Double	255	1.00000		9.01880	0.98120		0.00039	MHz
hpNormFcp5.filt	High Pass		Hamming		Double	45	0.50000		0.49395	0.50605		0.03682	Norm
lp150fc2p5.filt	Low Pass	150.00000	Chebyshev	8.00000	Double	255	2.50000		2.21800	2.21800		0.00000	MHz
lp20fc2cheby4.filt	Low Pass	20.00000	Chebyshev	4.00000	Double	255	2.00000		2.03987	2.03987		0.00048	MHz
lp20fc3p2.filt	Low Pass	20.00000	Chebyshev	4.00000	Double	255	3.20000		3.21691	3.21691		0.00047	MHz
lp20fc5.filt	Low Pass	20.00000	Blackman		Double	57	5.00000		4.91316	4.91316		0.00233	MHz
lp20fc5kaiser3.filt	Low Pass	20.00000	Kaiser	3.00000	Double	255	5.00000		5.03373	5.03373		0.04026	MHz
lp300fc25.filt	Low Pass	300.00000	Chebyshev	4.50000	Double	255	25.00000		24.67335	24.67335		0.00013	MHz
lp300fc75cheby3p5.filt	Low Pass	300.00000	Chebyshev	3.50000	Double	255	75.00000		75.36811	75.36811		0.00163	MHz
lp50fc3.filt	Low Pass	50.00000	Rectangular		20-bit	39	3.00000		3.00741	3.00741		0.86073	MHz

Figure 5.2 - The Filter Inspector

5.3 Export FIR Filter Coefficients

The FIR filter coefficients of any fixed precision filter design may be exported as a CSV (comma separated value) file and then used by a FPGA design. Before exporting filter coefficients, the Engineer should first test the filter design using either the Fixed-Rate Filter Model (section 7.1) or the Multirate Filter Model (section 7.2).

Figure 5.3 below illustrates the Export FIR Coefficients window after the user selected the low pass filter “lp50fc3” and clicked on the “Export Selection” button.

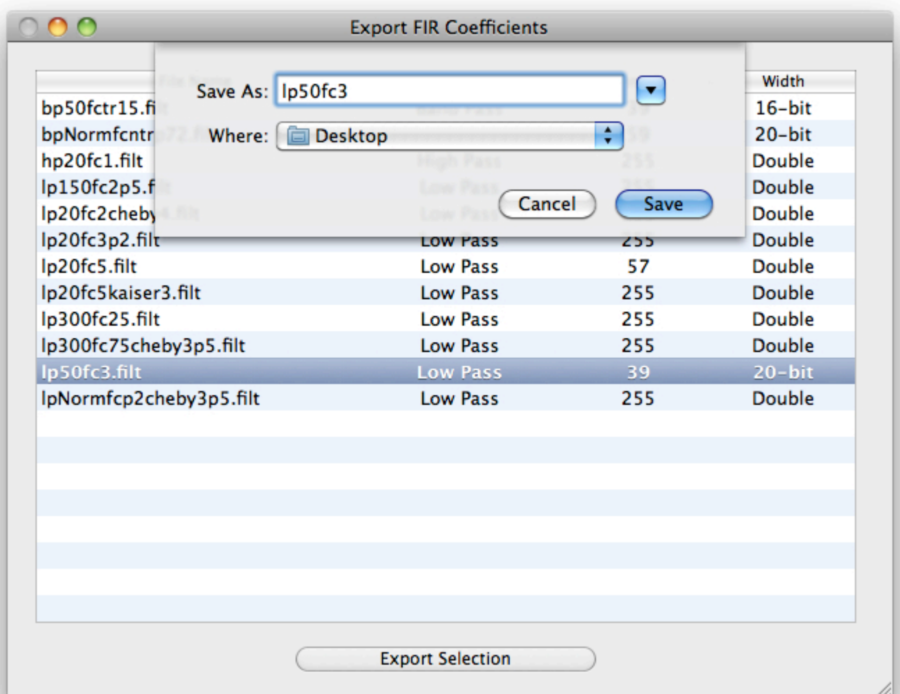


Figure 5.3 - The Export FIR Coefficients window

6.0 Signal Menu

The Signal menu's items allow the user to generate test signals for use with the models of sections 7.1 and 7.2, and to view the parameters of each previously generated signal via the Signal Inspector.

6.1 Generate Signal

An example of the Signal Generator window is illustrated in figure 6.1. The figure shows both the time and frequency domain graphs for a discrete signal x[n] that is composed of integral multiples of a 10 MHz analog frequency with a sampling rate of 300 MHz. The signal was sampled such that its maximum frequency component was oversampled at a 10x rate.

The filename of the signal, "Cos10M3Fs300" is indicated in the titlebar of the figure.

6.1.1 Sampling Rate

The Sampling rate for a waveform is set via the "Fs" input field as shown in figure 6.1. The sampling rate field must match the sampling rate of the filter designed in section 5.1 in order to be utilized by either the fixed-rate (section 7.1) or multirate (section 7.2) models.

6.1.2 Function Select

The function select view is located directly under the time domain graph as shown in figure 6.1. The stepper control is located adjacent to the function select, and may be incremented or decremented to select either a sinusoidal or miscellaneous function as shown in tables 6.1.2.1 and 6.1.2.2 respectively.

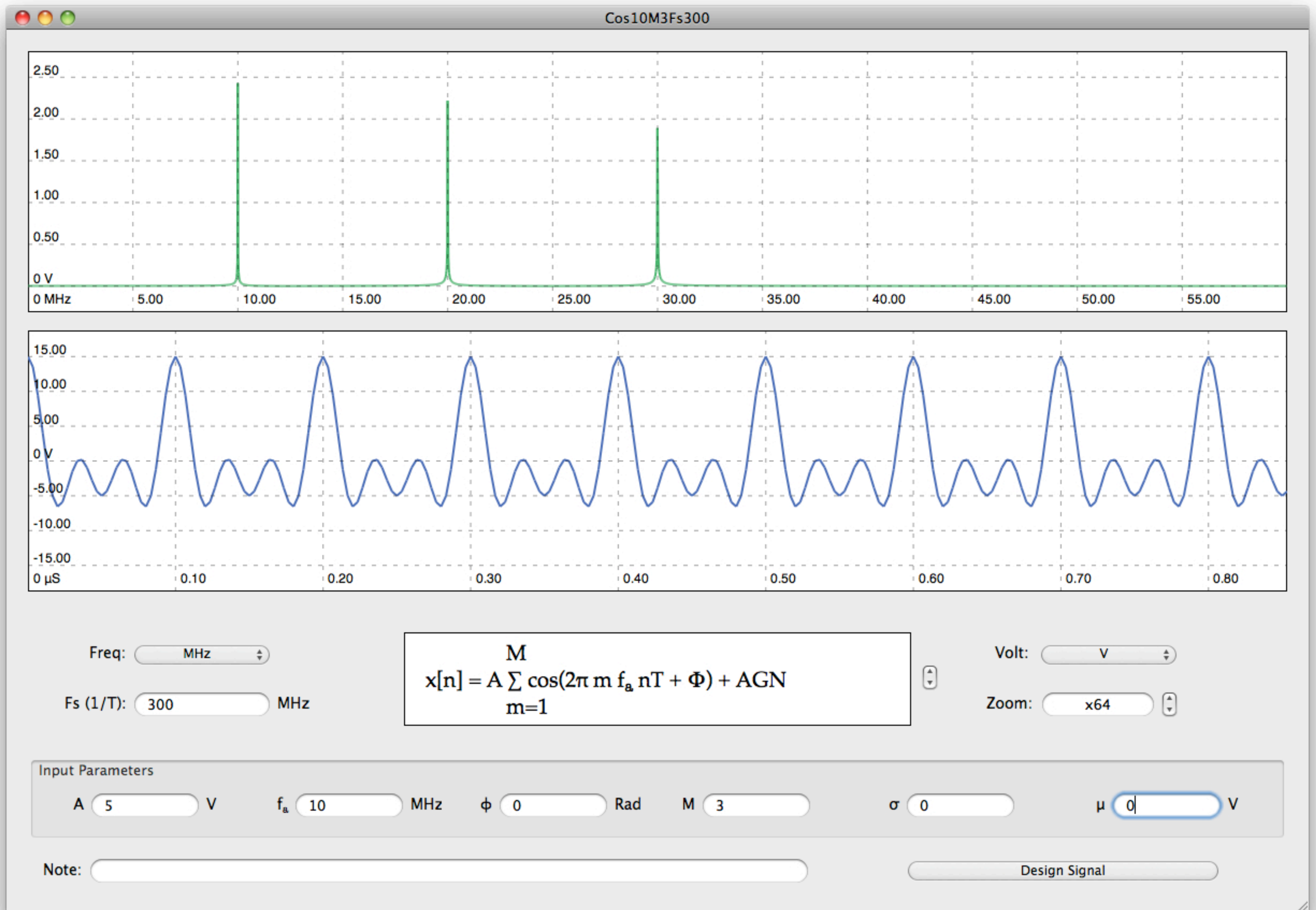


Figure 6.1 - The Signal Generator Window

Sinusoidal Function	Amplitude	Analog Frequency	Phase	Count
$x[n] = A \cos(2\pi f_a nT + \Phi)$	A	f_a	Φ	
$x[n] = A_1 \cos(2\pi f_{a1} nT) + A_2 \cos(2\pi f_{a2} nT)$	A_1, A_2	f_{a1}, f_{a2}		
$x[n] = A \sum_{m=1}^{M} \cos(2\pi m f_a nT + \Phi)$	A	f_a	Φ	M
$x[n] = A \sin(2\pi f_a nT + \Phi)$	A	f_a	Φ	
$x[n] = A_1 \sin(2\pi f_{a1} nT) + A_2 \sin(2\pi f_{a2} nT)$	A_1, A_2	f_{a1}, f_{a2}		
$x[n] = A \sum_{m=1}^{M} \sin(2\pi m f_a nT + \Phi)$	A	f_a	Φ	M
$x[n] = A_1 \sin(2\pi f_{a1} nT) * A_2 \sin(2\pi f_{a2} nT)$	A_1, A_2	f_{a1}, f_{a2}		

Table 6.1.2.1 - The Signal Generator's Sinusoidal Functions

Miscellaneous Function	Mean	Standard Deviation	Tau	Beta
$x[n] = \mu + \text{Rayleigh}(\sigma) \cos()$	μ	σ		
$x[n] = A\, \text{rect}(nT/\tau)$			τ	
$x[n] = \text{sinc}(\beta \pi nT)$				β
$x[n] = \text{sinc}^2(\beta \pi nT)$				β

Table 6.1.2.2 - The Signal Generator's Miscellaneous Functions

The sinusoidal functions of table 6.1.2.1 consist of three fundamental types: a single sinusoid at an analog frequency, a sum of two sinusoids of different analog frequencies, and a summation of ‘M’ sinusoids that are integral multiples of an analog frequency. In addition, the following mixer function is also provided:

$$x[n] = A_1 \sin(2\pi f_{a1} nT) * A_2 \sin(2\pi f_{a2} nT)$$

which produces the following result:

$$x[n] = \frac{A_1 A_2}{2} \cos(2\pi (f_{a1} - f_{a2}) nT) + \frac{A_1 A_2}{2} \cos(2\pi (f_{a1} + f_{a2}) nT)$$

In a strict mathematical sense the second term is negative, but since:

$$\cos(\theta) = - \cos(\theta)$$

the discrete waveform above is also mathematically correct. If the output of the mixer feeds a transmitter, the waveform is a form of AM modulation referred to as double-sideband, suppressed-carrier. The sum and difference terms in the equation are then referred to as the upper and lower sidebands respectively.

The miscellaneous functions of table 6.1.2.2 include gaussian noise, a rectangle function, and the sinc and sinc^2 functions. The gaussian noise is covered in detail in section 6.1.3, therefore is not discussed here. The rectangle, sinc, and sinc^2 functions were implemented as odd functions, namely:

$$x[n] \neq x[-n]$$

Since Fourier transforms of odd real functions produce complex results, the following is true:

$$\text{Re}\{ F(A\, \text{rect}(nT/\tau)) \} = \text{sinc}[n]$$

$$\text{Re}\{ F(\text{sinc}(\beta\, \pi\, nT) \} = \text{rect}[n]$$

$$\text{Re}\{ F(\text{sinc}^2(\beta\, \pi\, nT) \} = \text{triangle}[n]$$

where Re{F( )} represents the real component of the Fourier transform. Since the FFTea application displays the magnitude rather than the real component:

$$F(A\, \text{rect}(nT/\tau)) = |\, \text{sinc}[n]\, |$$

the rectangle function will display the absolute value of the sinc function.

6.1.3 Additive Gaussian Noise

Since random numbers with a Gaussian probability distribution cannot easily be generated from its corresponding probability density function, the Rayleigh probability distribution function can be used to generate Gaussian random variables. If R represents a Rayleigh distributed random variable with a probability distribution function:

$$F(R) = \begin{cases} 0 & R < 0 \\ 1 - e^{-R^2/2\sigma^2} & R \geq 0 \end{cases}$$

A Gaussian random variable G with mean μ can be written as:

$$G = \mu + R \cos(2 \pi r_2)$$

where r_2 is uniformly distributed over $0 \leq r_2 \leq 1$. Solving F(R) for R, the Rayleigh distributed random variable:

$$R(\sigma) = \sigma \sqrt{2 \ln(1/(1-r_1))}$$

where σ is the standard deviation, and r_1 is uniformly distributed over $0 \leq r_1 \leq 1$. Therefore, the Gaussian random variable can be defined as:

$$G(\sigma, \mu) = \mu + \sigma \sqrt{2 \ln(1/(1-r_1))} \cos(2 \pi r_2)$$

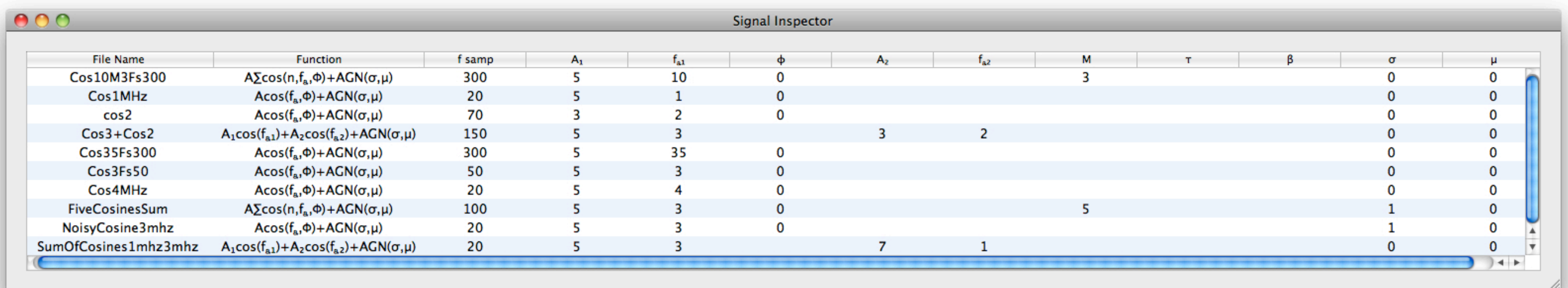
All functions in both tables 6.1.2.1 and 6.1.2.2 have an optional additive Gaussian noise parameter AGN as shown by the Signal Generator window of figure 6.1.

6.1.4 Time & Frequency Domain Graphs

The Signal Designer window illustrated in figure 6.1 shows both the time and frequency domain graphs for the selected function. The frequency graph will auto-scale such that the maximum frequency is optimally near the mid-range of its graph. The time graph has a zoom control with a stepper to manually magnify the x-axis as desired.

6.2 Signal Inspector

The Signal Inspector window summarizes all parameters of each generated signal in a tabular view as shown in figure 6.2. By default, the Signal Inspector is positioned at the lower left corner of the monitor upon the start of the FFTea application. The Inspector serves as a reference for both the Fixed-Rate Filter Model (section 7.1) and the Multirate Filter Model (section 7.2).



Signal Inspector

File Name	Function	f samp	A_1	f_{a1}	ϕ	A_2	f_{a2}	M	τ	β	σ	μ
Cos10M3Fs300	$A\sum\cos(n,f_a,\Phi)+AGN(\sigma,\mu)$	300	5	10	0			3			0	0
Cos1MHz	$A\cos(f_a,\Phi)+AGN(\sigma,\mu)$	20	5	1	0						0	0
cos2	$A\cos(f_a,\Phi)+AGN(\sigma,\mu)$	70	3	2	0						0	0
Cos3+Cos2	$A_1\cos(f_{a1})+A_2\cos(f_{a2})+AGN(\sigma,\mu)$	150	5	3		3	2				0	0
Cos35Fs300	$A\cos(f_a,\Phi)+AGN(\sigma,\mu)$	300	5	35	0						0	0
Cos3Fs50	$A\cos(f_a,\Phi)+AGN(\sigma,\mu)$	50	5	3	0						0	0
Cos4MHz	$A\cos(f_a,\Phi)+AGN(\sigma,\mu)$	20	5	4	0						0	0
FiveCosinesSum	$A\sum\cos(n,f_a,\Phi)+AGN(\sigma,\mu)$	100	5	3	0			5			1	0
NoisyCosine3mhz	$A\cos(f_a,\Phi)+AGN(\sigma,\mu)$	20	5	3	0						1	0
SumOfCosines1mhz3mhz	$A_1\cos(f_{a1})+A_2\cos(f_{a2})+AGN(\sigma,\mu)$	20	5	3		7	1				0	0

Figure 6.2 - The Signal Inspector

7.0 Model Menu

The Model menu contains both the fixed-rate "Sum & Filter" and Multirate models for simulating a FIR filter design from section 5.1.

7.1 Fixed-Rate "Sum & Filter" Model

The fixed-rate "Sum & Filter" model is used for systems or sub-systems with a single sampling rate. In order to properly test a FIR filter design (section 5.1) with generated signals (section 6.1) the sampling rate of the FIR filter and the test signals must be identical. The only exception is if the FIR filter design used normalized units, since in this case the filter can be used with any given sampling rate. (See section 5.1.4. for the use of normalized units).

Figure 7.1 illustrates the fixed-rate "Sum & Filter" model. Note that the filename of the model, "Cos10M3andCos35" is indicated in the titlebar of the figure.

7.1.1 Sum & Filter Model Diagram

A system block diagram of the fixed-rate model is shown below the time domain graph in figure 7.1. On the left of the diagram, two discrete signals $x_1[n]$ and $x_2[n]$ are input to the system, each with an identical sampling rate of f_s. These two discrete signals are then summed together as x[n], and input to the filter H[n] (shown in light blue). The filter H[n] outputs the discrete signal y[n]. Note that both y[n] and x[n] are shown to have an identical sampling rate of f_s, which is the same rate as that of the filter.

7.1.2 Input Signal Selection

Both the $x_1[n]$ and $x_2[n]$ discrete signals (referenced in the preceding section) have an associated and labeled pop-up menu that allows the user to select previously generated signals as input. Referring to figure 7.1, the $x_1[n]$ pop-up is set to "Cos10M3Fs300"; the $x_2[n]$ pop-up set to "Cos35Fs300". The discrete signal "Cos10M3Fs300" was previously illustrated in figure 6.1, and represents the summation of three cosines of analog frequencies 10 MHz, 20 MHz, and 30 MHz as shown. Note that the sampling rate in figure 6.1 is 300 MHz, which is the sampling rate of the example shown in figure 7.1.

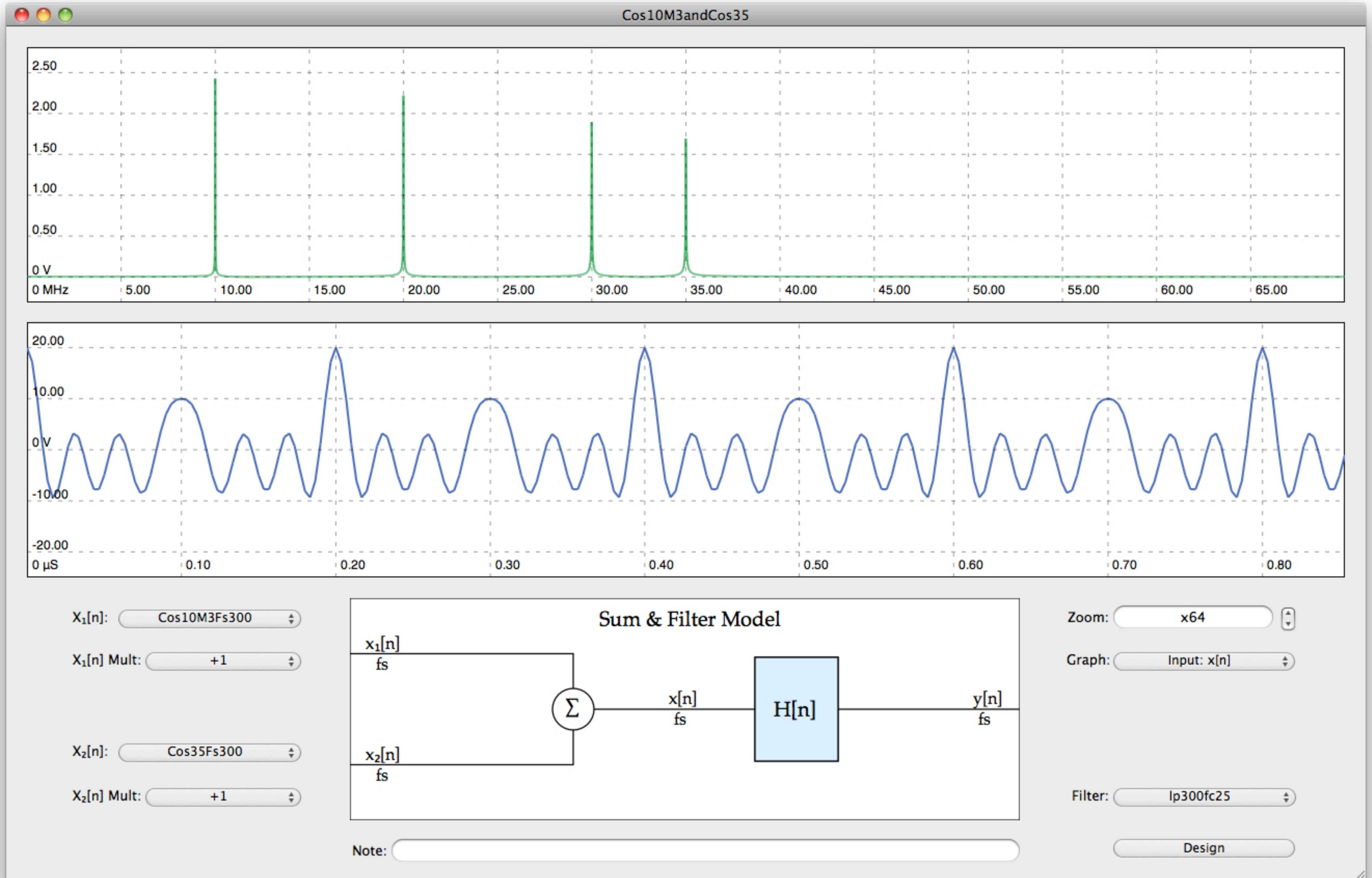


Figure 7.1 - The Fixed-Rate "Sum & Filter" Model Displaying Input Signal x[n]

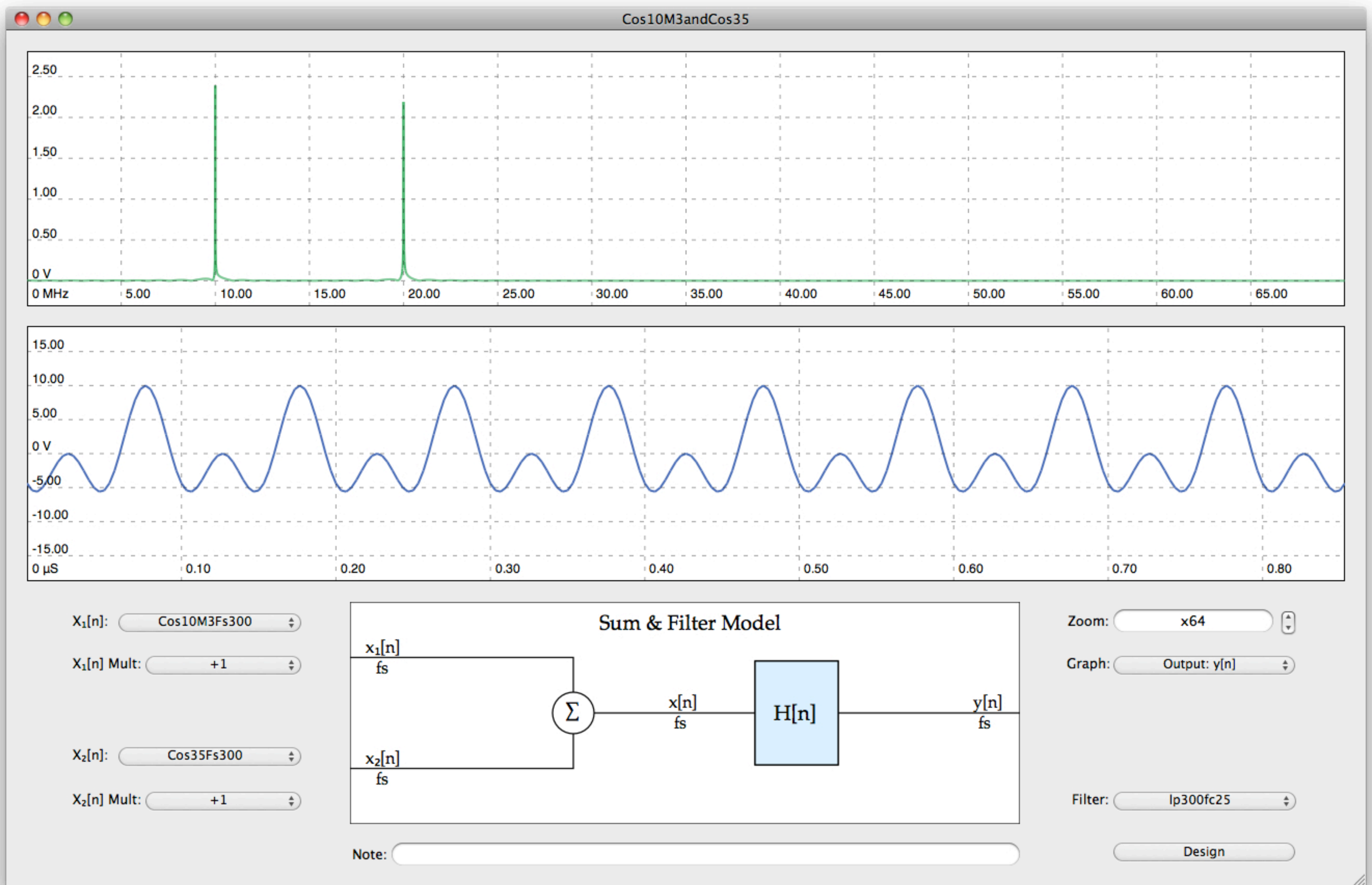


Figure 7.1.4 - The Fixed-Rate “Sum & Filter” Model Displaying Output Signal y[n]

The “Cos35Fs300” signal is a single cosine at an analog frequency of 35 MHz, and a sampling rate of 300 MHz.

Two additional pop-up menus labeled “x_1[n] Mult” and “x_2[n] Mult” are multipliers for discrete signals x_1[n] and x_2[n] respectively. These multipliers are used to change the sign of its corresponding input signal to either a positive “+1” or negative “-1” value. A third setting of “0” is available to inhibit its associated input signal, and is useful when the user requires only one discrete input signal rather than two. Note that both multipliers are set to “+1” in the example shown in figure 7.1.

As a result of the input signal selection, it is evident that the x[n] signal is the sum of four cosines at analog frequencies 10 MHz, 20 MHz, 30 MHz, and 35 MHz.

7.1.3 Filter Selection

The H[n] filter of section 7.1.1 has an associated pop-up menu that allows the user to select a previously designed filter for use in the model. Referring to figure 7.1, the filter pop-up is set to “lp300fc25”, which is the previously designed low pass filter shown in figures 5.1.1 and 5.1.2. The filter has a cutoff frequency of 25 MHz, and a sampling rate of 300 MHz.

7.1.4 Time & Frequency Domain Graphs

The fixed-rate “Sum & Filter” model illustrated in both figures 7.1 and 7.1.4 shows both a time and frequency domain graph that is similar to that of the Signal Generator window as shown in figure 6.1. The frequency graph will auto-scale such that the maximum frequency is optimally near the mid-range of its graph. The time graph has a zoom control with a stepper to manually magnify the x-axis as desired.

7.1.5 Input & Output Graph Set Selection

There are two sets of graphs for the model, the first set to display the discrete input signal x[n], and the second set for the discrete output signal y[n]. A pop-up menu labeled “Graph” allows the user to select either the system input or output as the source for both the time and frequency domain graphs as discussed in the prior section.

To view the system input, the graph pop-up is set to the input signal x[n] as illustrated in figure 7.1. As previously discussed in section 7.1.2, the input signal is composed of four cosines with analog frequencies of 10 MHz, 20 MHz, 30 MHz, and 35 MHz. The frequency domain graph clearly shows the summation of cosines at their proper frequencies. The time domain graph has a period of 100 nS, which corresponds to the lowest frequency component of 10 MHz.

To view the system output response, the graph pop-up is set to the output signal y[n] as illustrated in figure 7.1.4. The discrete output signal y[n] is equal to the convolution of the discrete input x[n] with the filter H[n]. As previously discussed in section 7.1.3, the filter is low pass, and has a cutoff frequency of 25 MHz. Consequently, the discrete output of the system y[n] is only composed of the 10 MHz and 20 MHz components as shown by the frequency domain graph. The time domain graph still has a period of 100nS, but is noticeably different due to filtering out the 30 MHz and 35 MHz components.

7.2 Multirate Filter Model

The multirate model is used for systems or sub-systems with an input sampling rate that is different from its output sampling rate. Multirate involves either interpolation (up-sampling), decimation (down-sampling) or a combination of both.

In order to properly test a multirate FIR filter design (section 5.1) with generated signals (section 6.1) the sampling rate of the FIR filter must be:

$$f_{s(\text{filter})} = L\, f_{s(\text{input})}$$

where L is the interpolation factor. The only exception is if the FIR filter design used normalized units, since in this case the filter can be used with any given sampling rate. (See section 5.1.4. for the use of normalized units).

Figure 7.2 illustrates the multirate model. Note that the filename of the model, “multirateUser300to450” is indicated in the titlebar of the figure.

7.2.1 Multirate Model Diagram

A system block diagram of the multirate model is shown below the time domain graph in figure 7.2. On the left of the diagram, two discrete signals $x_1[n]$ and $x_2[n]$ are input to the system, each with an identical sampling rate of f_s. These two discrete signals are then summed together as x[n], and input to an interpolator (shown in light green). The interpolator increases the sampling rate by a factor of L, such that the sampling rate at the output of the interpolator is Lf_s for the discrete signal w[m]. The filter H[m] (shown in light blue) has input w[m] and output p[m] sampled at the same Lf_s rate. The discrete signal p[m] is input to a decimator (shown in light green) which decreases the sampling rate by a factor of M. The output of the decimator is a new discrete sequence y[k] with a sampling rate of:

$$f_{s(\text{out})} = \frac{L\, f_s}{M}$$

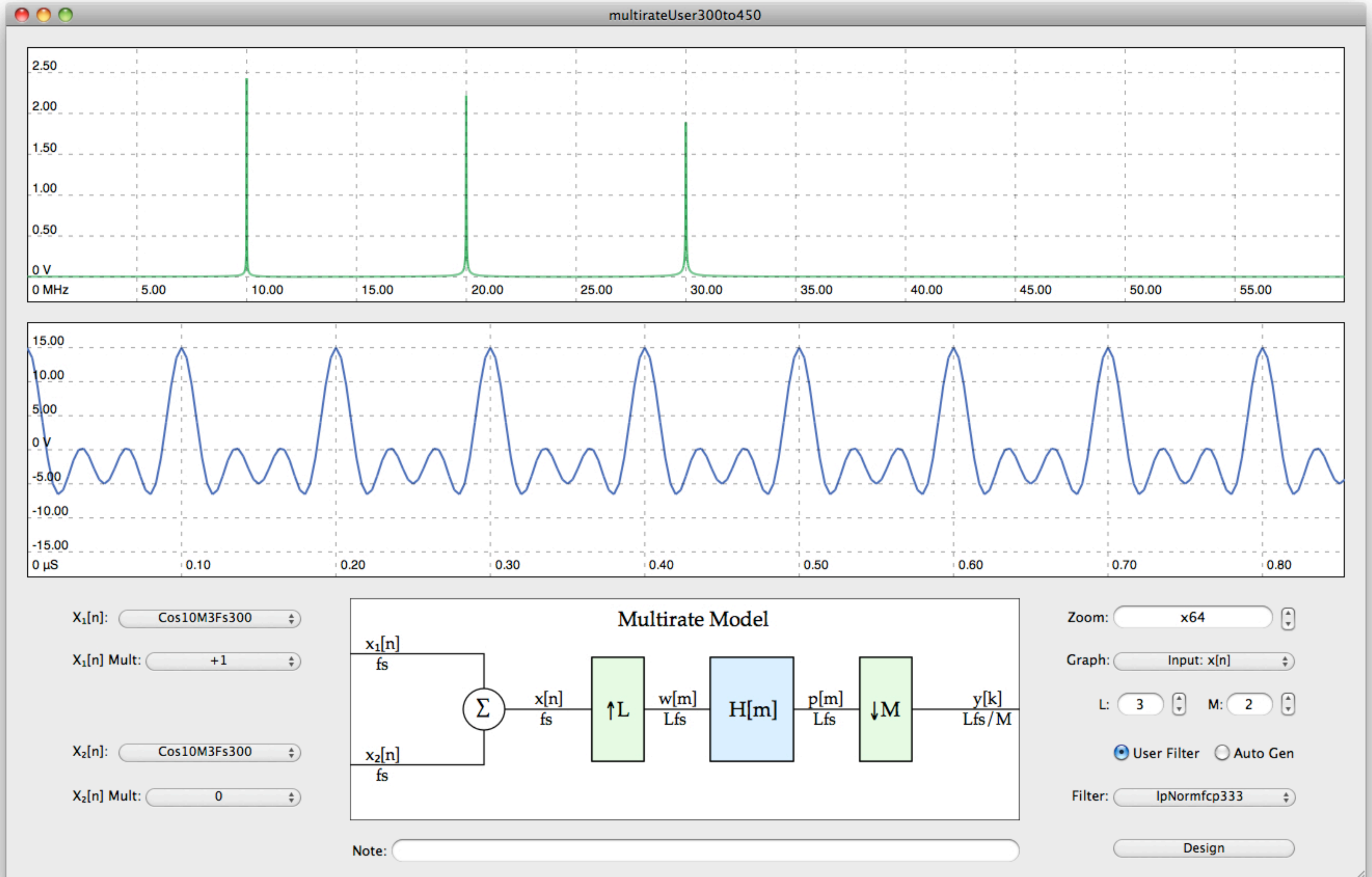


Figure 7.2 - The Multirate Model Displaying Input Signal x[n]

7.2.2 Input Signal Selection

The method of input signal selection for the multirate filter model is identical to that of the fixed-rate model of section 7.1.2.

Referring to figure 7.2, the sole input signal is $x_1[n]$, since the multiplier for $x_2[n]$ is set to zero. The pop-up for input signal $x_1[n]$ is set to “Cos10M3Fs300”, which was previously illustrated in figure 6.1, and represents the summation of three cosines of analog frequencies 10 MHz, 20 MHz, and 30 MHz as shown. Note that the sampling rate in figure 6.1 is 300 MHz, which is the sampling rate of the example shown in figure 7.2.

7.2.3 Interpolation (L) & Decimation (M)

The interpolation (L) and decimation (M) values are individually set using steppers as shown in figure 7.2. Each stepper may either increment or decrement its associated text field located adjacent to the stepper. The range of each stepper is the set of the integers from 1 to 10.

As noted in section 7.2.2, the input frequency is 300 MHz. Figure 7.2 identifies L=3 and M=2, therefore the output frequency is 450 MHz (using the equation of section 7.2.1).

7.2.4 Filter Selection

Filter selection can be achieved either by selecting a previously designed filter from section 5.1, or optionally having a filter auto-generated with the correct specifications. Referring to figure 7.2, a radio button located below the interpolation and decimation fields allows the user to optionally choose an auto-generated filter design.

7.2.4.1 User-Designed Multirate Filter

Referring to figures 7.2 and 7.2.4.1, the filter pop-up is set to “lpNormfcp333”, which is a previously designed low pass filter with a cutoff frequency of 1/3 in normalized units. The simplest approach to designing a multirate filter H[m] is as follows:

- Use a Low Pass Filter with Normalized Units. (The use of normalized units are less susceptible to user error since the filter is independent of sampling frequency.)

- Set the Cutoff Frequency of the Low Pass Filter to 1/M if M > L, or 1/L if L > M.

Since L > M in the example of figure 7.2, the cutoff frequency is set to 1/L or 1/3.

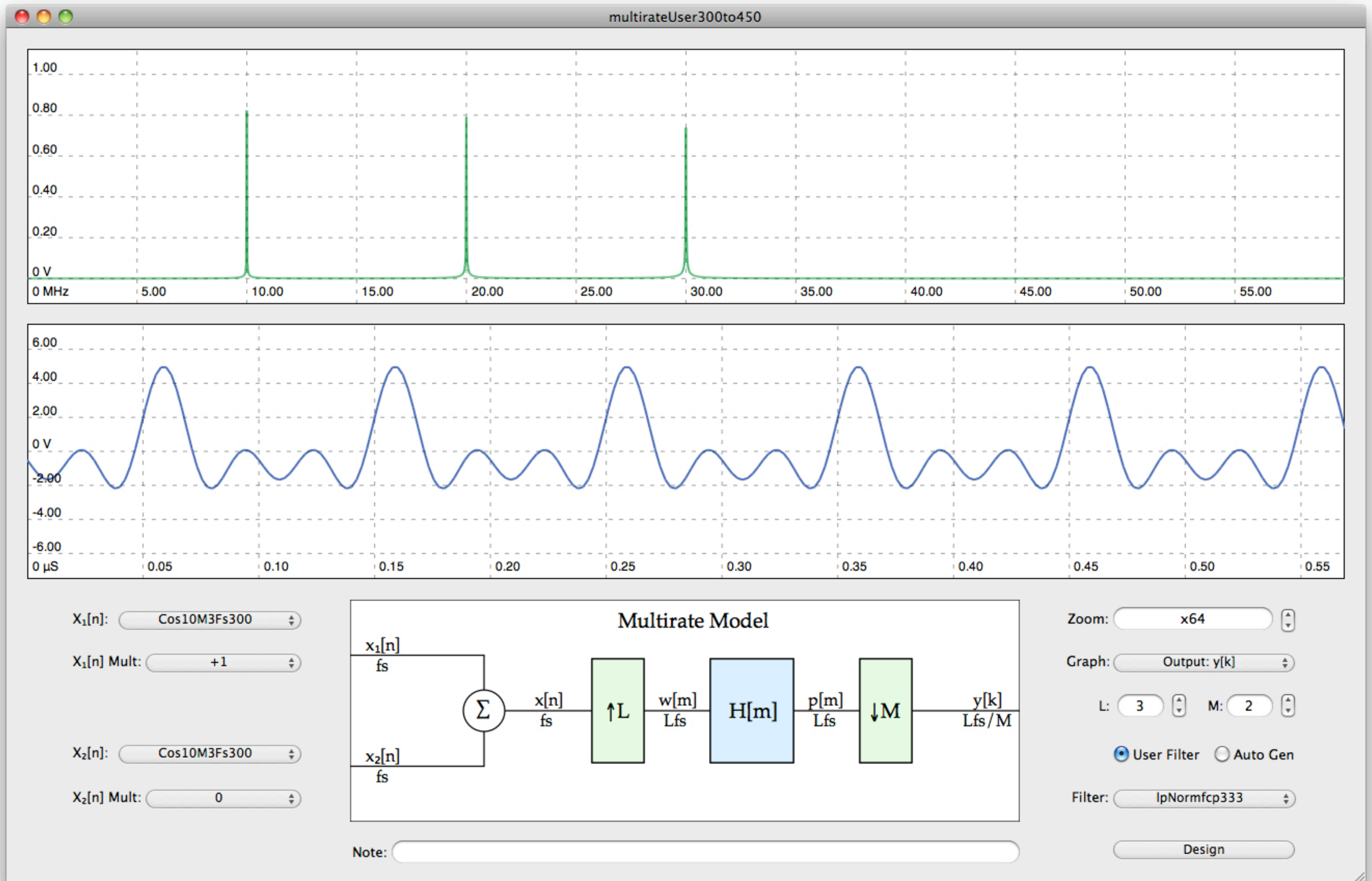


Figure 7.2.4.1 - The Multirate Model Displaying Output Signal y[k]

7.2.4.2 Auto-Generated Multirate Filter

Figure 7.2.4.2 illustrates the multirate filter model using an auto-generated filter. Note that the radio button is set to “Auto-Gen” and the filter pop-up menu is not visible. The auto-generated multirate filter is designed using the same procedure as outlined in section 7.2.4.1. The auto-generated filter is useful to test against a user designed filter.

7.2.5 Time & Frequency Domain Graphs

The multirate model illustrated in both figures 7.2 and 7.2.4.1 shows both a time and frequency domain graph that is similar to that of the Signal Generator window as shown in figure 6.1. The frequency graph will auto-scale such that the maximum frequency is optimally near the mid-range of its graph. The time graph has a zoom control with a stepper to manually magnify the x-axis as desired.

7.2.6 Input & Output Graph Set Selection

There are two sets of graphs for the model, the first set to display the discrete input signal x[n], and the second set for the discrete output signal y[k]. Note that the input and output are a function of different discrete sequences since x[n] and y[k] have different sampling rates.

To view the system input, the graph pop-up is set to the input signal x[n] as illustrated in figure 7.2. The discrete input signal x[n] was designed using the signal generator as illustrated in figure 6.1, and is equivalent to the summation of three cosines of analog frequencies 10 MHz, 20 MHz, and 30 MHz sampled at a rate of 300 MHz. The frequency domain graph clearly shows the cosines at their proper frequencies. The time domain graph has a period of 100 nS, which corresponds to the lowest frequency component of 10 MHz.

To view the system output response, the graph pop-up is set to the output signal y[k] as illustrated in figure 7.2.4.1 Note that the frequency domain graph of output signal y[k] has identical frequency components to the discrete input signal x[n], but with different magnitudes. The magnitudes of the output are 1/3 the magnitudes of the input. Similarly, the time domain graph of output signal y[k] has the same waveform as the discrete input signal x[n], but the amplitude is reduced by 1/3 at the output. The reason the output has a gain of 1/3 is due to the interpolation process. The interpolation factor L will always determine the gain of the output signal y[k]. In the example illustrated L=3; hence 1/3.

Figure 7.2.4.2 illustrates the system output response using an automatically generated multirate filter as described in section 7.2.4.2. Note that both the time and frequency domain responses are identical to the user designed filter response of figure 7.2.4.1.

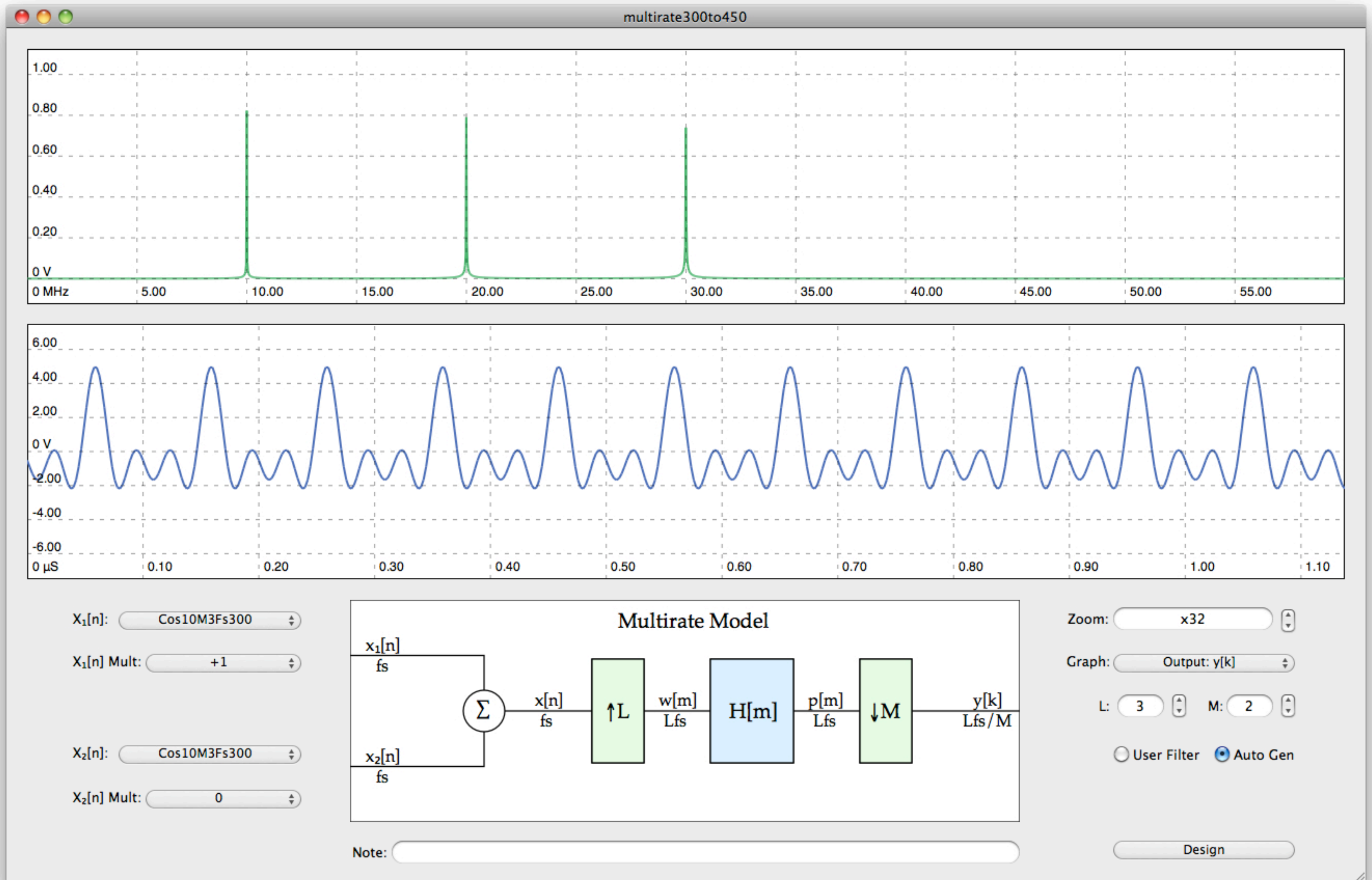


Figure 7.2.4.2 - The Multirate Model Displaying Output Signal y[k] using an Auto-Generated Filter

8.0 Revision History

Revision	Date	Details
1.0	5 Nov 2010	Initial release.

fftea.com