



Graphical User Interface Design of FIR Filter

Myagmardorj Bayasgalan, Xiang-E Sun

College of Electronic & Information, Yangtze University, Jingzhou, China

Email: baysaa0709@yahoo.com

How to cite this paper: Bayasgalan, M. and Sun, X.-E. (2018) Graphical User Interface Design of FIR Filter. *Open Access Library Journal*, 5: e4709.

<https://doi.org/10.4236/oalib.1104709>

Received: June 4, 2018

Accepted: December 25, 2018

Published: December 28, 2018

Copyright © 2018 by authors and Open Access Library Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

FIR digital filter is a stable linear phase filter. Relying on the powerful computing power of MATLAB software and its GUI platform, this graduation project has completed the GUI design and implementation of FIR digital filter through programming. Through the FIR filter to achieve the program, you can design and implement a variety of amplitude characteristics of the FIR digital filter, according to the user input filter design indicators, select the corresponding filter design method (window function, frequency sampling method) to design the corresponding filter. According to the different amplitude characteristics, the parameters of the filter design parameters will change accordingly; when the Kaiser window design filter is selected, the user is allowed to input the stopband attenuation of the filter; when using the frequency sampling method to design the filter, the user must input the filter length and the transition zone sample value; when the parameter is entered incorrectly, the system will prompt for an error message. GUI can also provide intuitive filter amplitude characteristics, phase characteristics and the unit impulse response graphics to help users to confirm whether the design of the filter meets the design specifications; by setting the global shortcut keys, the user after all parameters and options is confirmed by the Shortcut keys to achieve the rapid implementation of filter design.

Subject Areas

Computer Engineering

Keywords

MATLAB, GUI , FIR Digital Filter, Design and Implementation

1. Introduction

1.1. Introduction of Digital Filter

The digital filter is a discrete system, and it can do a series of mathematic

processing to the input signal, and therefore obtain the desired information from the input signal. There are many kinds of digital filters, and also many different ways to classify them [1]. According to their function, the FIR filters can be classified into four categories, which are lowpass filter, highpass filter, bandpass filter, and bandstop filter. According to the impulse response, there are usually two types of digital filters, which are finite impulse response (FIR) filters and infinite impulse response (IIR) filters. According to the formula above, if a_i is always zero, then it is a FIR filter. Otherwise, if there is at least one non-zero a_i , then it is an IIR filter. Usually, we need three basic arithmetic units to design a digital filter, which are the adder, the delay, and the multiplier.

1.2. FIR Filter

The finite impulse response (FIR) filter is one of the most basic elements in a digital signal processing system, and it can guarantee a strict linear phase frequency characteristic with any kind of amplitude frequency characteristic. Besides, the unit impulse response is finite; therefore, FIR filters are stable system. Process of designing the FIR filter is as shown in **Figure 1**. The FIR filter has a broad application in many fields, such as telecommunication, image processing, and so on.

The system function of FIR filter is

$$H(z) = \sum_{k=0}^{L-1} h(k) z^{-k} \quad (1)$$

where L is the length of the filter, and $h(k)$ is the impulse response.

2. Methods of FIR Digital Filters

2.1. Window Function Method

We will design filters that meet our requirements based on our own ideas. In general, we begin by studying the filter problem from the filter's frequency response function $H_d(w)$. Filter frequency response function expression can be written

$$H_d(w) = \sum_{n=-\infty}^{\infty} h_d(n) e^{-jwn} \quad (2)$$

$h_d(n)$ is the unit impulse response of $H_d(w)$. If we can know $H_d(w)$ we can find $h_d(n)$ through expression (2),

$$h_d(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(w) e^{jwn} dw \quad (3)$$

If $h_d(n)$ can be known, the frequency response function of the filter can be calculated by (1), and Z-transform can be used to find the system function.

In general, unit sample response $h_d(n)$ obtained from the above relation is infinite in duration, so it must be truncated at some point say $n = M - 1$ to yield an FIR filter of length M (i.e. 0 to $M - 1$). This truncation of $h_d(n)$ to length $M - 1$ is same as multiplying $h_d(n)$ by the rectangular window defined as

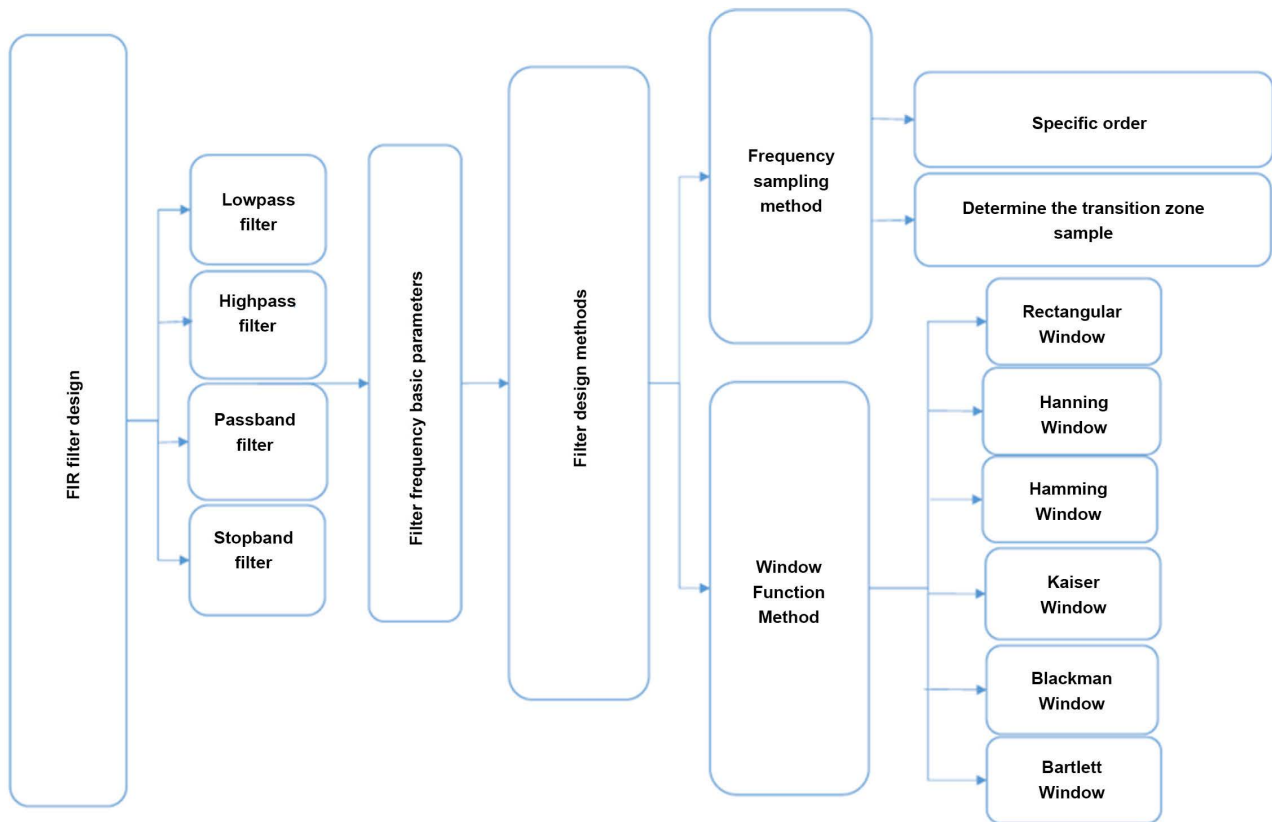


Figure 1. FIR filter design.

$$w(n) = \begin{cases} 1 & 0 \leq n \leq M-1 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Thus the unit sample response of the FIR filter becomes

$$h(n) = \begin{cases} hd(n) & w(n) \\ hd(n) & 0 \leq n \leq M-1 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Now, the multiplication of the window function $w(n)$ with $h_d(n)$ is equivalent to convolution of $H_d(w)$ with $W(w)$, where $W(w)$ is the frequency domain representation of the window function

$$W(w) = \sum_{n=0}^{M-1} w(n) e^{-jwn} \quad (6)$$

Thus the convolution of $H_d(w)$ with $W(w)$ yields the frequency response of the truncated FIR filter

$$H(w) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(v) W(w-v) dv \quad (7)$$

The frequency response can also be obtained using the following relation

$$H(w) = \sum_{n=0}^{M-1} h(n) e^{-jwn} \quad (8)$$

Thus the frequency response obtained by using (7) contains ripples in the frequency domain. In order to reduce the ripples, instead of multiplying $h_d(n)$ with a rectangular window $w(n)$, $h_d(n)$ is multiplied with a window function that contains a taper and decays toward zero gradually, instead of abruptly as it occurs in a rectangular window. As multiplication of sequences $h_d(n)$ and $w(n)$ in time domain is equivalent to convolution of $H_d(w)$ and $W(w)$ in the frequency domain, it has the effect of smoothing $H_d(w)$.

The several effects of windowing the Fourier coefficients of the filter on the result of the frequency response of the filter are as follows:

- 1) A major effect is that discontinuities in $H(w)$ become transition bands between values on either side of the discontinuity.
- 2) The width of the transition bands depends on the width of the main lobe of the frequency response of the window function, $w(n)$ *i.e.* $W(w)$.
- 3) Since the filter frequency response is obtained via a convolution relation, it is clear that the resulting filters are never optimal in any sense.
- 4) As M (the length of the window function) increases, the mainlobe width of $W(w)$ is reduced which reduces the width of the transition band, but this also introduces more ripple in the frequency response.
- 5) The window function eliminates the ringing effects at the bandedge and does result in lower sidelobes at the expense of an increase in the width of the transition band of the filter.

This article will introduce six typical window functions, which are Rectangle Window, Bartlett Window, Hanning, Hamming, Blackman, Kaiser Window [2].

1) Rectangle Window

$$W_R(n) = R_N(n) \quad (9)$$

2) Bartlett Window

$$W_B(n) = \begin{cases} \frac{2n}{N-1} & 0 \leq n \leq 0.5(N-1) \\ 2 - \frac{2n}{N-1} & 0.5(N-1) < n \leq N-1 \end{cases} \quad (10)$$

3) Hanning window

$$W_{Hn}(n) = 0.5 \left[1 - \cos\left(\frac{2\pi n}{N-1}\right) \right] R_N(n) \quad (11)$$

4) Hamming

$$W_{Hm}(n) = \left[0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \right] R_N(n) \quad (12)$$

5) Blackman

$$W_{Bl}(n) = \left(0.42 - 0.5 \cos\frac{2\pi n}{N-1} + 0.08 \cos\frac{4\pi n}{N-1} \right) R_N(n) \quad (13)$$

Figure 2 shows the plots of the five classic windows in the time domain is:

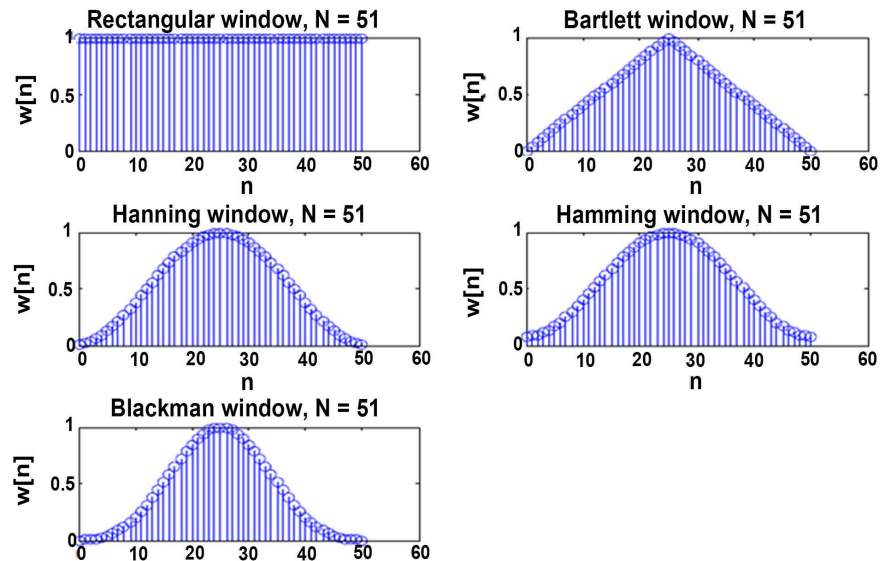


Figure 2. Plots of the five classic windows in the time domain.

6) Kaiser Window

The Kaiser window has some differences from the other five window functions. The other five window functions are fixed at design time, so the sidelobe amplitude is also fixed in each case. However, the parameters of the Kaiser window can be adjusted, which is an optimized function.

$$W_k(n) = \frac{I_0(\beta)}{I_0(\alpha)}, \quad 0 \leq n \leq N-1 \quad (14)$$

2.2. Frequency Sampling Method

In this method, the given frequency response is sampled at a set of equally spaced frequencies to obtain N samples. Thus, sampling the continuous frequency response $H_c(w)$ at N points essentially gives us the N -point DFT of $H_d(2\pi nk/N)$.

Thus by using the IDFT formula, the filter coefficients can be calculated using the following formula

$$h(n) = \frac{1}{N} \sum_{k=0}^{N-1} H(k) e^{-j(2\pi n/N)k} \quad (15)$$

Now using the above N -point filter response, the continuous frequency response is calculated as an interpolation of the sampled frequency response. The approximation error would then be exactly zero at the sampling frequencies and would be finite in frequencies between them. The smoother the frequency response being approximated, the smaller will be the error of interpolation between the sample points [3] [4].

There are two different set of frequencies that can be used for taking the samples. One set of frequency samples are at $fk = k/N$ where $k = 0, 1, \dots, N-1$. The other set of uniformly spaced frequency samples can be taken at

$$fk = (k+1/2)/N \text{ for } k = 0, 1, \dots, N-1.$$

Merits of frequency sampling method

1) Unlike the window method, this technique can be used for any given magnitude response.

2) This method is useful for the design of non-prototype filters where the desired magnitude response can take any irregular shape. There are some disadvantages with this method *i.e.* the frequency response obtained by interpolation is equal to the desired frequency response only at the sampled points. At the other points, there will be a finite error present.

3. The Realization of Fir Filter with Matlab

a) The Merits of Matlab GUI

GUI is communication platform between user and computer. By using GUI, user can conveniently communicate with Matlab program and control program. Matlab offers ample GUI design instructions which make the realization of GUI more convenient, intuitive and flexible [5].

FIR digital filter design needs to achieve the task

Since the FIR filter is designed in MATLAB GUI, we can design according to FIR filter [6].

The process selectively designs filters. The tasks required to design the FIR filter are as follows:

1) The filter selection box is fixed as FIR filter. The program can be developed by follow-up personnel to design the IIR filter Wave device.

2) In the response characteristic selection of the filter, you can choose low-pass filter, high-pass filter, band-pass filter Wave filter, band-rejection filter and four filter response types.

3) Input the sampling frequency and the passband of the filter in the basic parameter input box of the filter. Frequency, filter stopband cutoff frequency, passband attenuation, stopband attenuation and other design criteria.

4) There is an input parameter error alarm mechanism; when the parameter input is not reasonable, it will pop up error input dialogue Box error message.

5) After selecting the response characteristics of the filter, three design methods for designing FIR filters are selected: window function method, frequency sampling method, and equal-ripple optimal approximation method.

6) When the choice of FIR filter window function design, you can choose rectangular window, triangular window, Hanning window, Hamming window, Blackman window, Kaiser window to design FIR filter; when the frequency sampling method is selected, FIR filter needs to be provided with specified order and transition sampling value; when selecting the equal ripple best approximation method, it is necessary to provide passband attenuation and stopband attenuation to design the FIR filter.

7) In the graphic display area, there are three coordinate axes. You can draw three maps, respectively, the frequency response diagram, phase frequency re-

sponse diagram and unit impulse response diagram.

8) The minimum order calculated by this method can be displayed above the unit impulse response axis.

9) Each case has a default parameter settings and global shortcut response when running in the UI interface by comparing and analyzing the amplitude characteristics of the filter shown in **Figure 1**, the amplitude characteristics of w on $[0, 2\pi]$ can be found different.

b) GUI of FIR filter

The GUI implementation of FIR filter: However, when using MATLAB FIR filter design, processes the data in $[0,1]$. The results of Window methods and Frequency sampling method at different Filters are represented in figures. **Figure 3** shows the realization of a low-pass filter rectangular window at the wf (Sampling frequency = 10000 Hz), $wp1$ (Passband cut off frequency) = 3000 Hz, $ws1$ (Stopband cut off frequency) = 3300 Hz. **Figure 4** shows the realization of a high-pass filter Kaiser window at the ($wf = 10000$ Hz, $wp1 = 3100$ Hz, $ws1 = 2600$ Hz). **Figure 5** shows the realization of a pass-band filter frequency sampling method at the ($T1 = 0.25$, $N = 60$, $wp1 = 3500$ Hz, $ws1 = 3000$ Hz, $as = 31$ dB). **Figure 6** shows the realization of a stop-band filter frequency sampling method at the ($T1 = 0.25$, $N=60$, $wp1 = 3000$ Hz, $ws1 = 3500$ Hz, $as = 29$ dB).

Analysis of the design features of the window function method:

When we use the window function method, we only need two parameters, one is the basic frequency parameter of the filter, and the other is the selection of the window function type. But in the design of Kaiser window, besides the above parameters, we need to provide the value of stopband attenuation. So if you want to fix the stopband attenuation value, you can design the FIR filter through the Kaiser window. If you want to design a FIR filter using only the base frequency parameters, you can use several other window functions, and we can choose the most appropriate window function by comparing the passband attenuation, the actual value of stopband attenuation and the value of the minimum order. After changing the filter parameters, we can analyze which window function is the best under the window function method in the same way.

Analysis of the design features of the frequency sampling method:

In the use of the frequency sampling method, three conditions are required. One is the basic frequency parameter of the filter, the specified order value, and the other is the given transition sampling value (a value between $[0,1]$). The influence of the filter on the stopband attenuation can be analyzed by adjusting the order of the filter and the size of the transition sampling value, respectively, and because of the characteristics of frequency sampling method, this method has many advantages over window function method, so the frequency sampling method rather than the window function method is generally chosen when the computational quantity is not taken into account and the minimum order is not required.

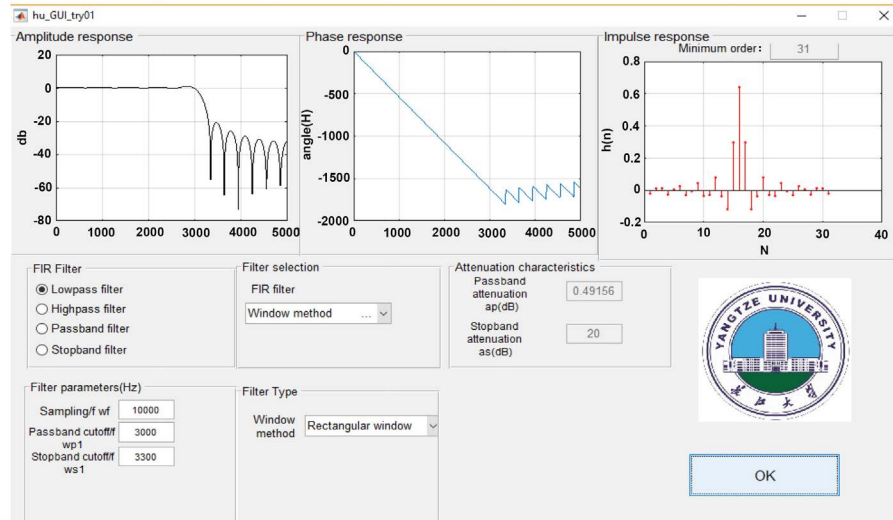


Figure 3. Realization of a low-pass filter rectangular window ($w_{p1} = 3000$ Hz, $w_{s1} = 3300$ Hz, $a_s = 20$ dB, minimum order 31).

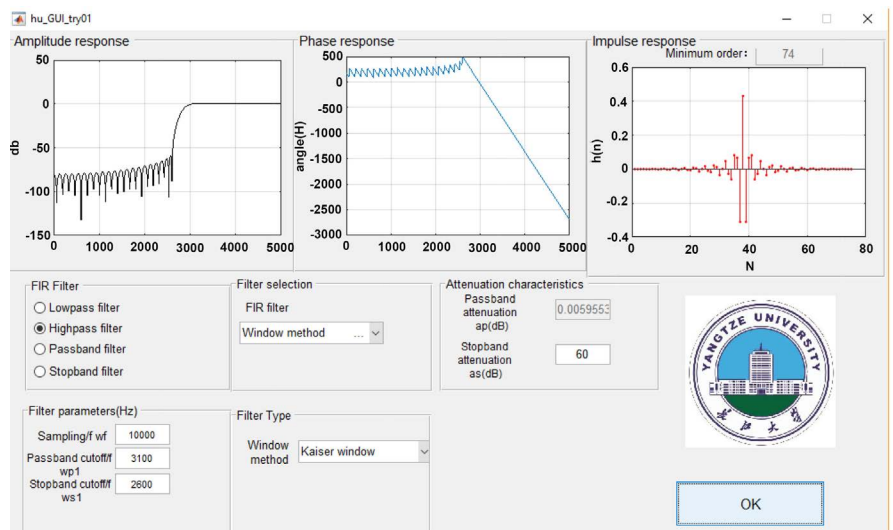


Figure 4. Implementation of a high-pass filter Kaiser window ($w_{p1} = 3100$ Hz, $w_{s1} = 2600$ Hz, $a_s = 60$ dB, minimum order 74).

4. Conclusions

The GUI realization of FIR filter was offered in the form of example in this paper. The main idea of the research includes its characteristics of linear phase, two methods of implementation and programming in MATLAB GUI.

The linear phase of the FIR filter can be further divided into the first type of linear phase and the second type of linear phase. In the frequency domain, the length N of the filter is combined to take the characteristics of even and odd numbers. The filter is divided into four cases to discuss its frequency domain characteristic. Next, the two methods of FIR digital filter implementation, and the characteristics of each design method and the advantages and disadvantages between them are analyzed in detail. The two design methods have been

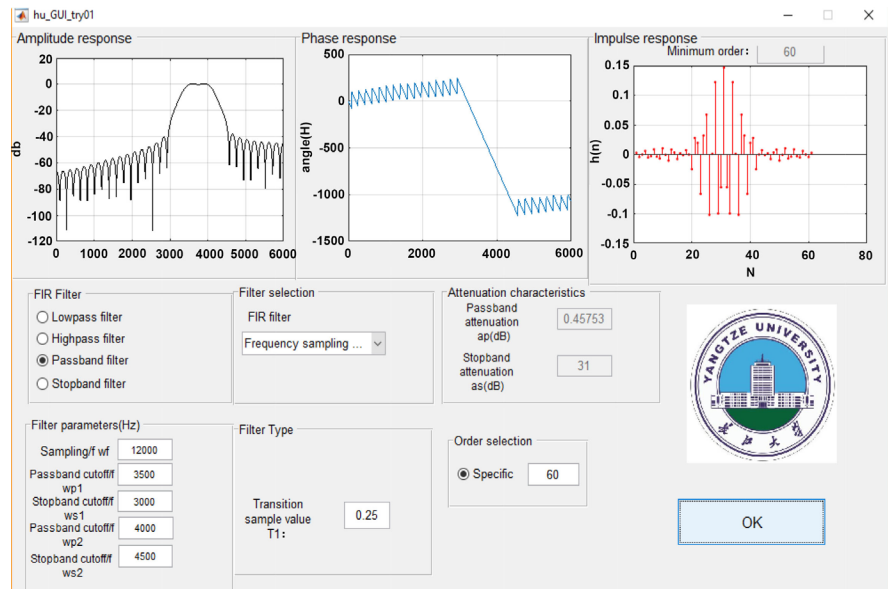


Figure 5. Realization of a pass-band filter frequency sampling method ($T1 = 0.25$, $N = 60$, $wp1 = 3500$ Hz, $ws1 = 3000$ Hz, $as = 31$ dB).

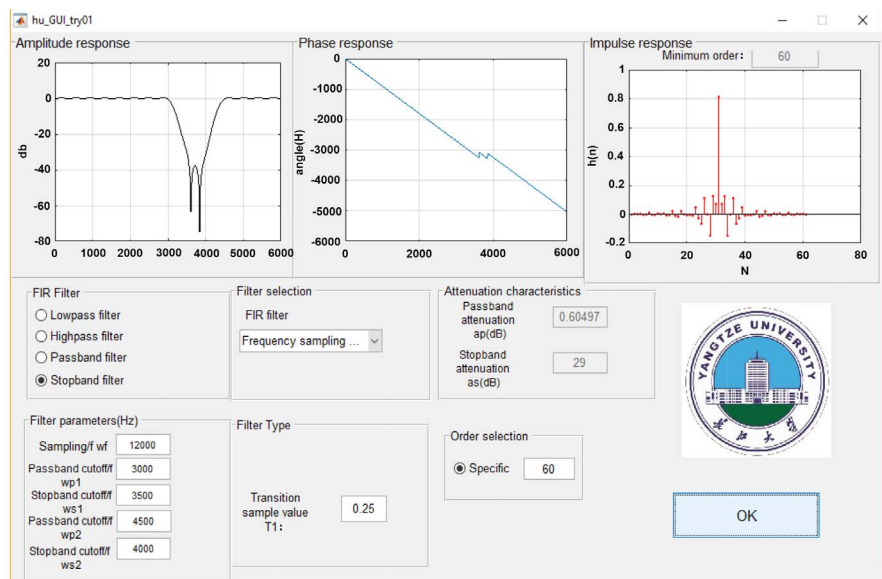


Figure 6. Realization of a stop-band filter frequency sampling method ($T1 = 0.25$, $N = 60$, $wp1 = 3000$ Hz, $ws1 = 3500$ Hz, $as = 29$ dB).

optimized so that each design method can be implemented: filter amplitude characteristics selection and input of basic parameters; passband attenuation, stopband attenuation and filter order can be displayed in real time. The filter can draw the amplitude frequency response, phase frequency response, and unit impulse response of the device can be accurately plotted. Because the input of the basic parameters of the filter is wrong, a special interaction design is made for the possible problems. For wrong data, the data will be checked before running. If the input data is incorrect, an error dialog box will appear.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Cheng, M. (2006) Analysis of Digital Filters Based on MATLAB. *Communication Power Supply Technology*, No. 1, 25-67.
- [2] Lyons, R.G. (2012) *Understanding Digital Signal Processing*. 3rd Edition, Information, No. 8.
- [3] Liu, W.S. (2008) *Design of FIR Digital Filter Based on DSP*. Guizhou University, Guiyang.
- [4] Saramaki, T. and Neuvo, Y. A Class of FIR Nyquist (Nth-Band) Filters with Zero Intersymbol Interference.
- [5] Wang, M.Y. and Zhong, W. (2007) *Construction and Application Based on GUI of Matlab*. Modern Electrical Power.
- [6] Chen, Y.Y. (2007) *Detail Comment of Signal Processing Based on Matlab*. People Post & Telecom Press, Beijing.