Lab 3

Report for Module EEP55C22 Computational MethodsJiacheng Li, Student ID 23330637

Trinity College Dublin

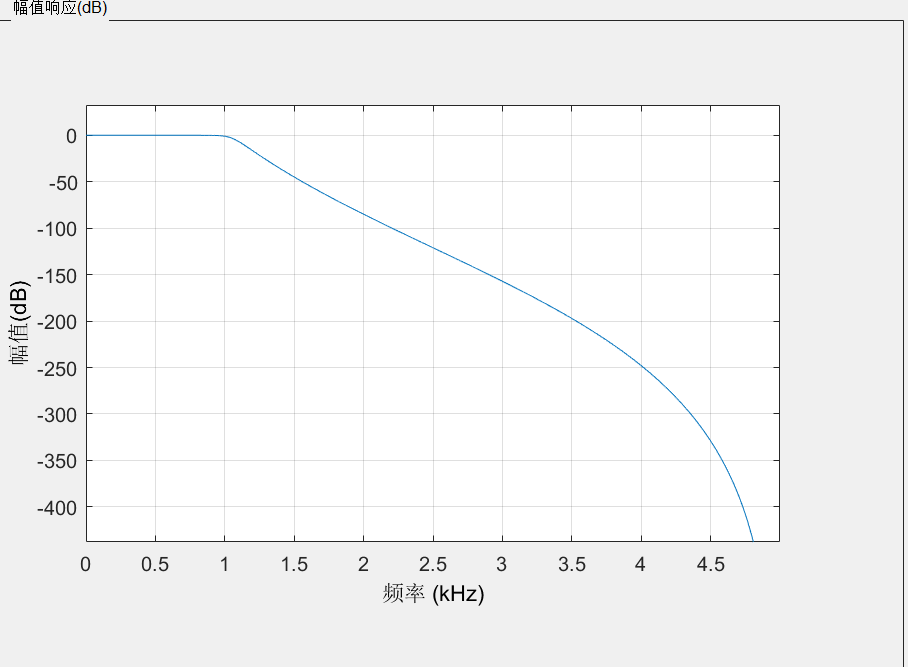
lij31@tcd.ie

October 2023

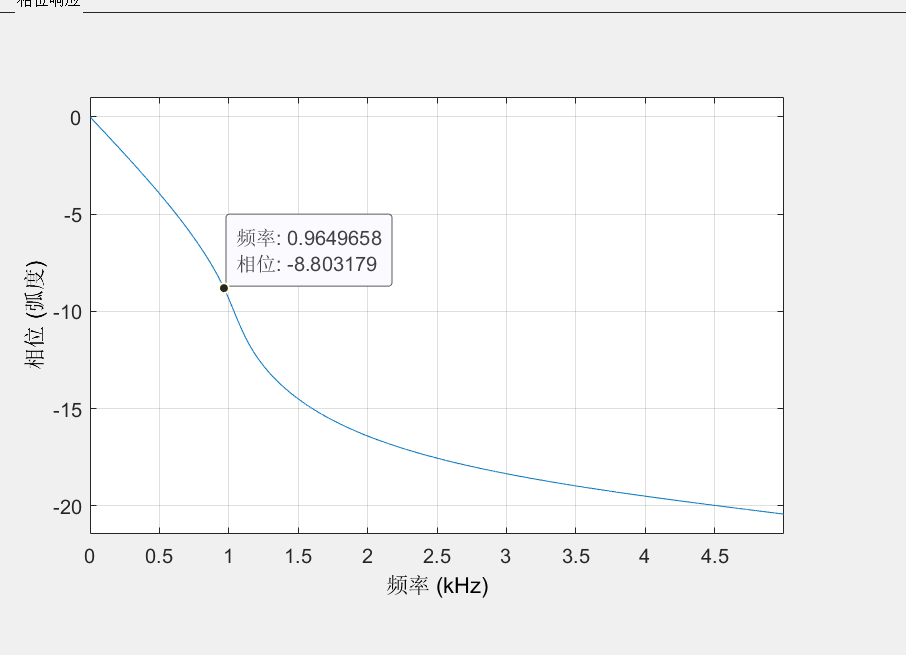
*This report is submitted in part fulfilment for the assessment required in EEP55C05 Digital Signal Processing. I have read and I understand the plagiarism provisions in the General Regulations of the University Calendar for the current year.*

1.1.1

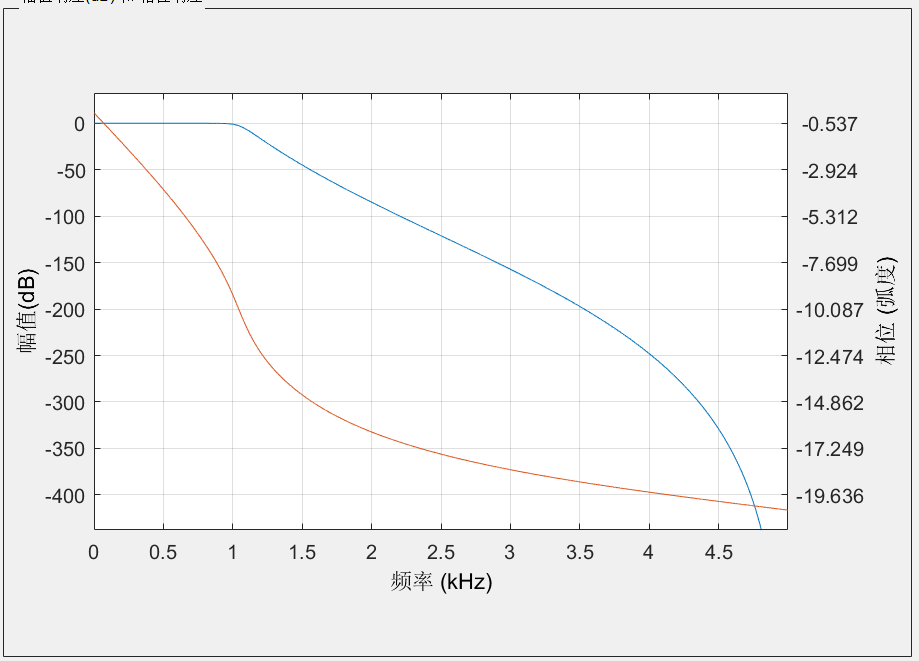
The picture of the magnitude responses shows:



The picture of the phase responses shows:



We can mix these two pictures:

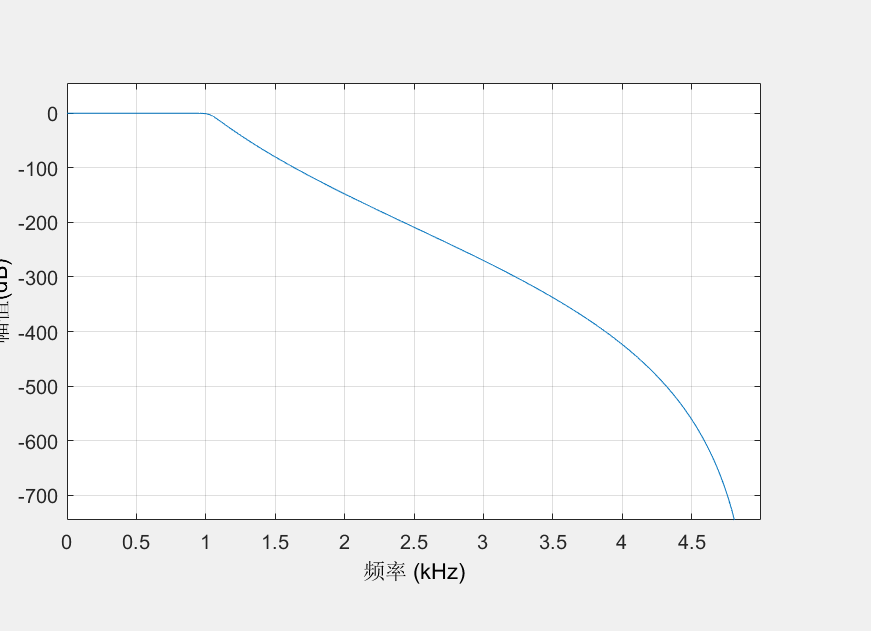


The order of the IIR filter is 13 the phase is actually what I expect for an IIR filter.

1.1.2

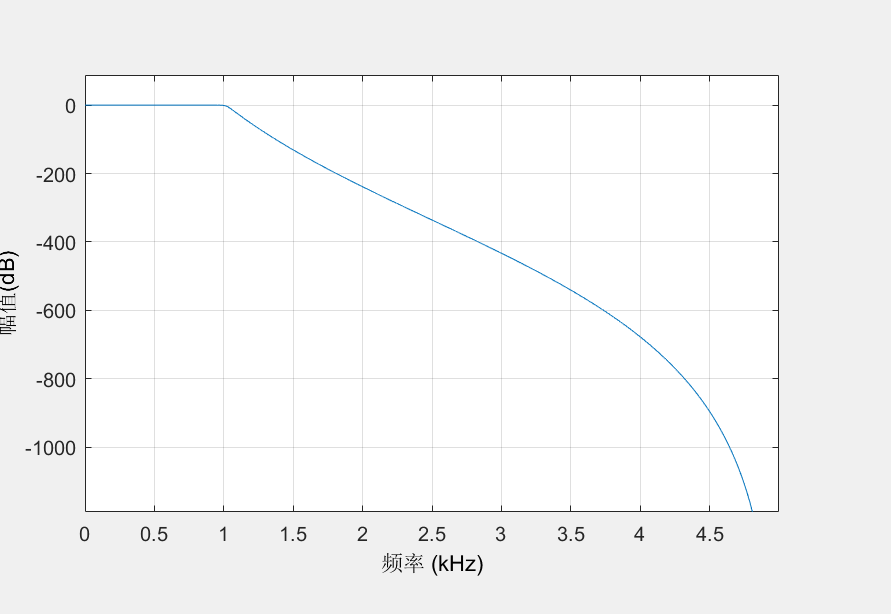
Narrow the transition region:

Change the Fstop: to 1.5kHz



And the order change to be 22

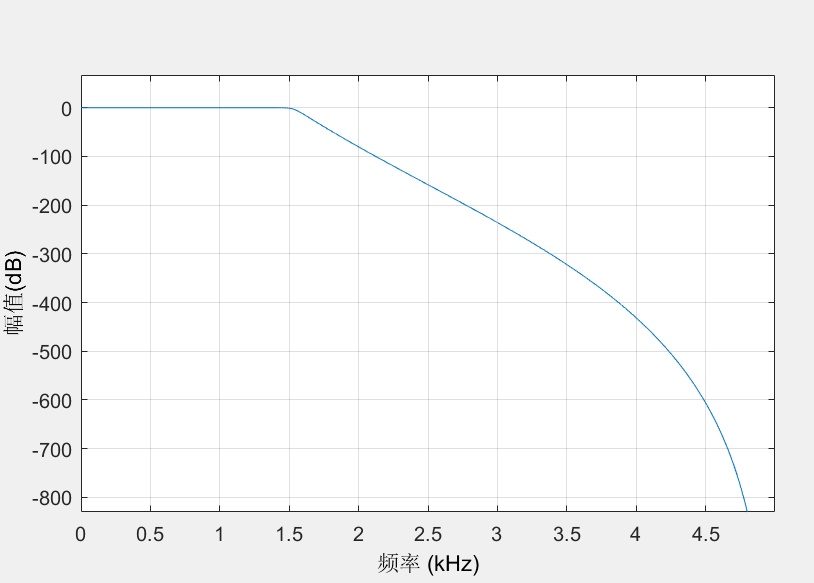
Change the Fstop: 1.3kHz:



The order change to be 35.

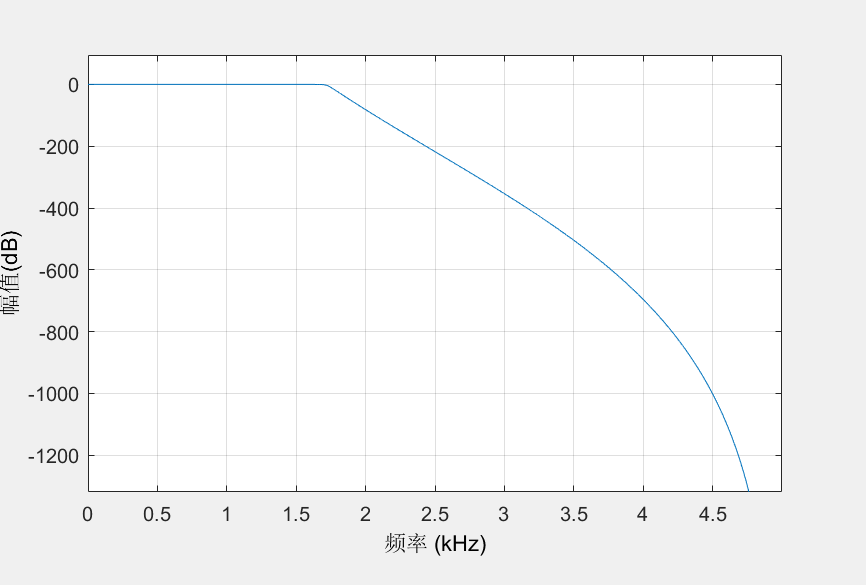
Keep the Fstop to 2 and change Fpass:

Change the Fpass to 1.5kHz:



The order change to be 28.

Change the Fpass to 1.7kHz:

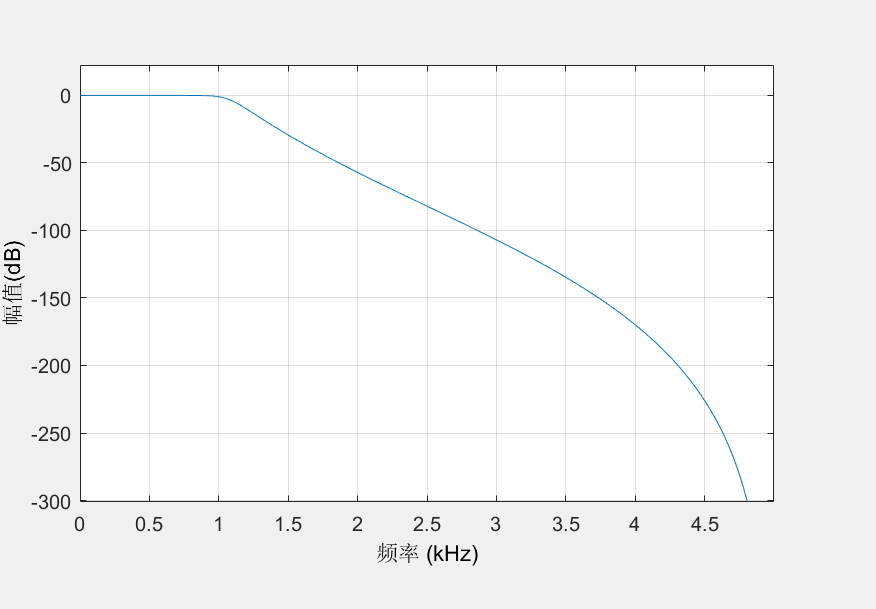


The order change to be 49.

So, it shows that when we narrow the transition region, the order will be higher.

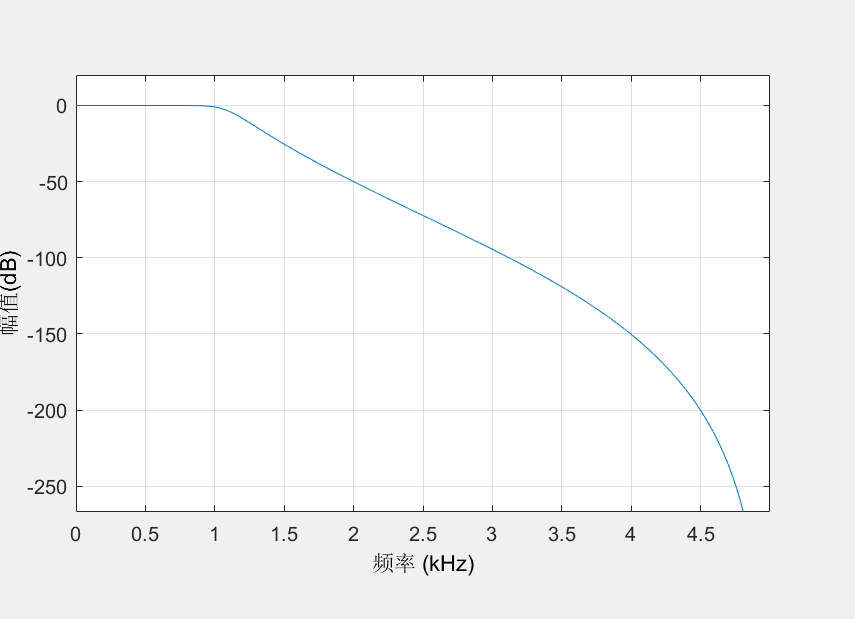
Widen the transition region:

Change the Fstop: to 2.5kHz



And the order change to be 9

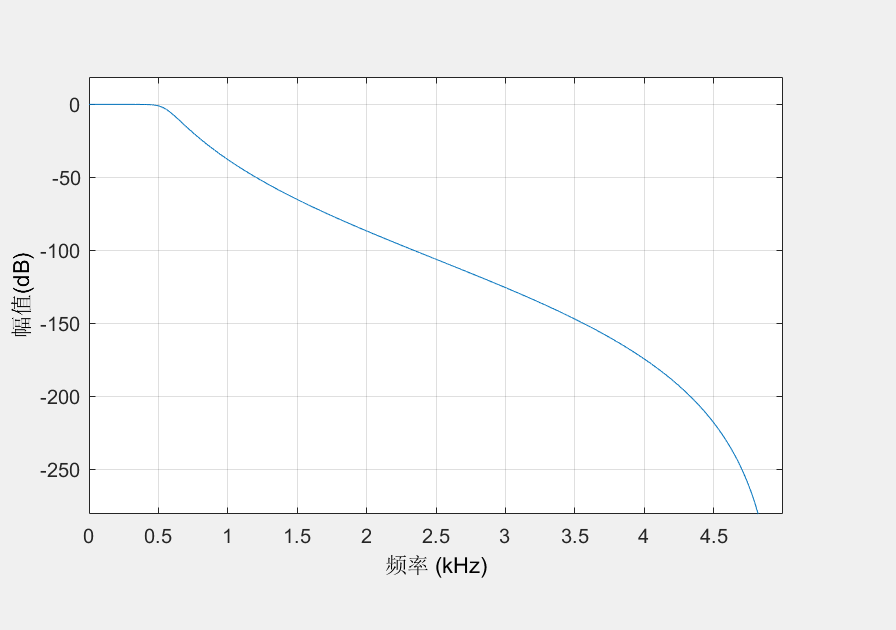
Change the Fstop: 2.7kHz:



The order change to be 8.

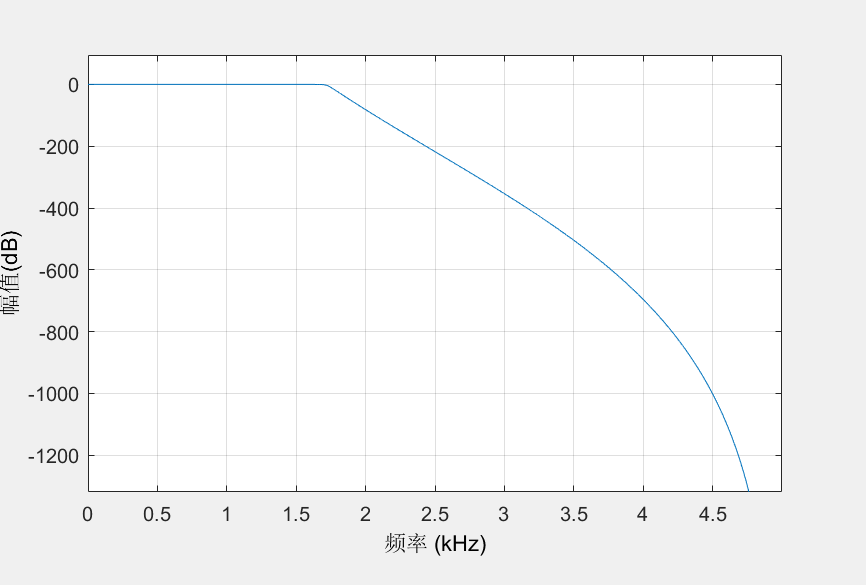
Keep the Fstop to 2 and change Fpass:

Change the Fpass to 0.5kHz:



The order change to be 7.

Change the Fpass to 0.3kHz:



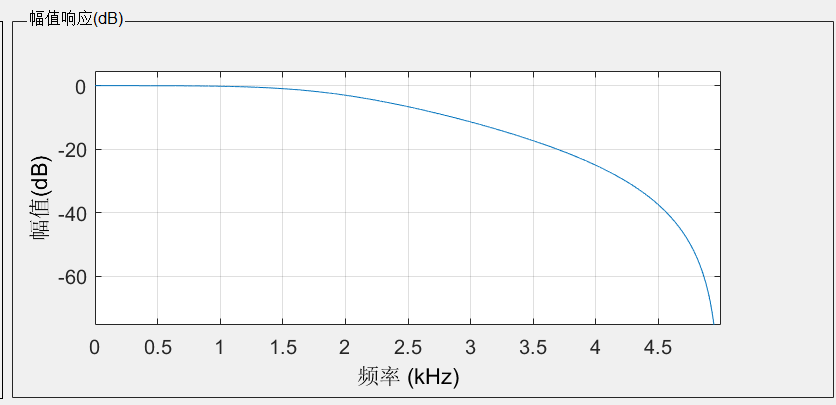
The order change to be 7.

So, it shows that when we widen the transition region, the order will be lower than before.

1.1.3

With its discrete-time filter we can calculate the order should be 1.64 Thus, the minimum order should be 2.

So when we choose Fs to be 10kHz and Fc to be 2kHz, the magnitude response will be shown like this:

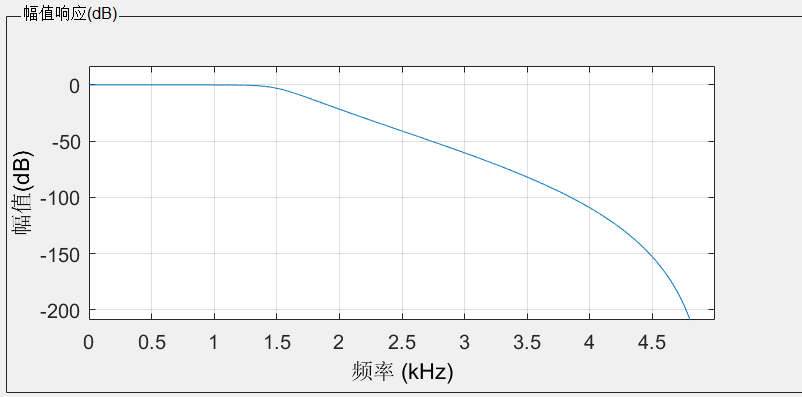


1.1.4

Since we choose the Fs = 10Khz as 1.1.1:

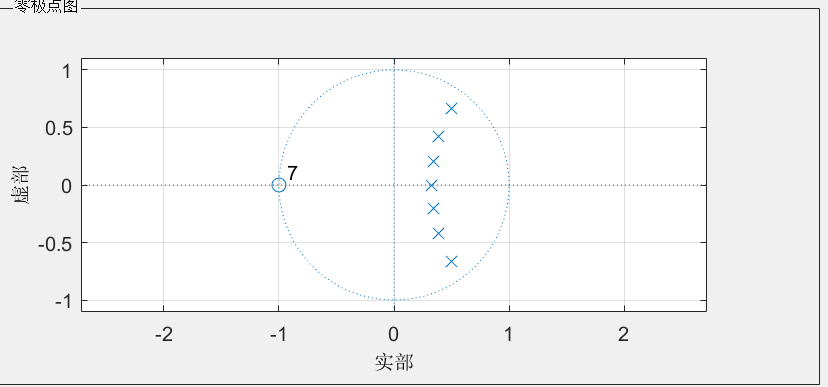
We need to fit the equation:

Which gives us that the order should be 7 so that conditions can be met. In this condition we get the picture:



From the picture it is obviously that the picture fits attenuation of 40dB at 3kHz and 3dB cutoff frequency.

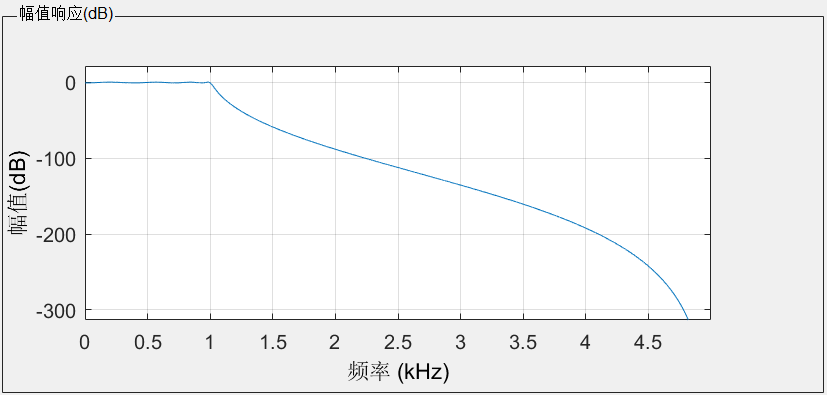
1.1.5



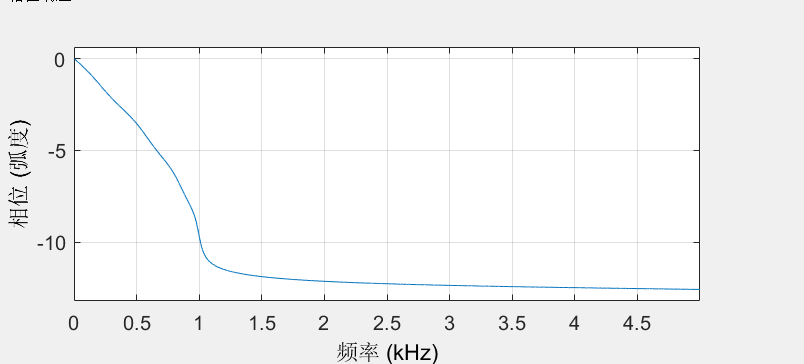
It doesn’t match the correspond to the poles that you can find with the impulse invariance method.

1.1.6

The picture of the magnitude responses shows:



The picture of the phase responses shows:



The order of the Chebyshev filter is 8

1.1.7

Pros:

Compared with the Butterworth filter. When these two kinds of filters have the same order, Chebyshev filter usually has a steeper roll down, which is more ideal at the cut-off frequency than Butterworth filter.

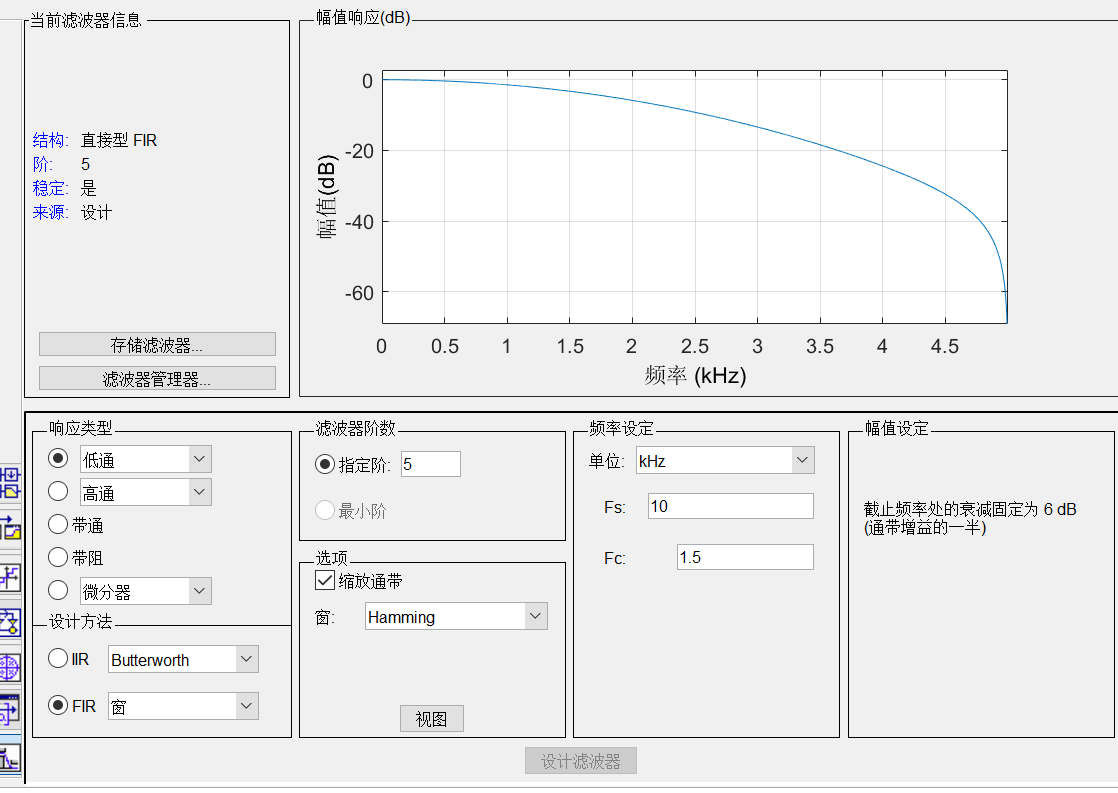
Cons：

Chebyshev filter has an equal amplitude fluctuation in the frequency response within the passband or stopband, while Butterworth is the flattest within the passband.

1.2.1

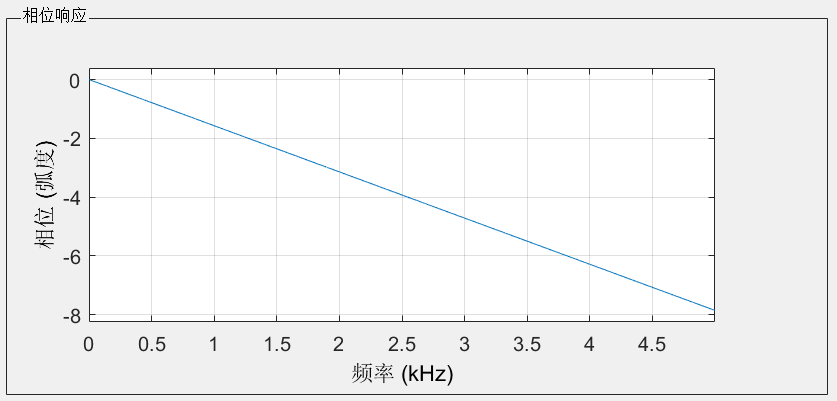
minimum order, stable, low-pass Butterworth filter with a passband frequency of 1 kHz

and a stopband frequency of 2 kHz. Assume a sampling frequency of 10 kHz. Make the attenuation 1 dB at the passband frequency and 80 dB at the stopband frequency.

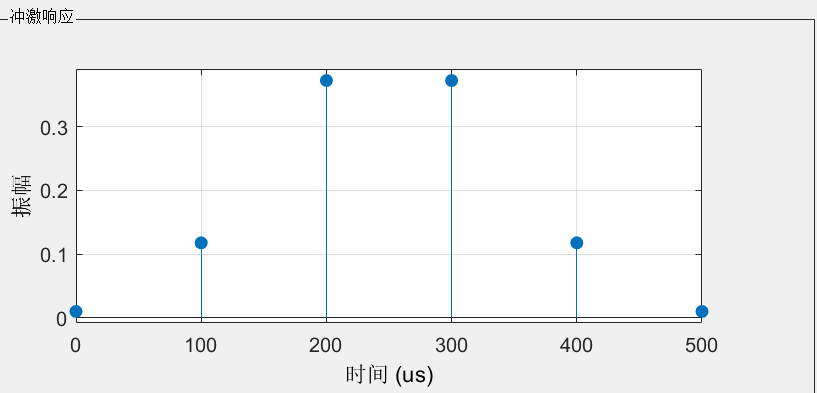


Phase differs from the IIR filter：In FIR filter, the amplitude reduces with an increasing reduced rate. And when the frequency is zero, the amplitude is near zero instead of being zero in IIR filter.

The picture of the phase responses shows:



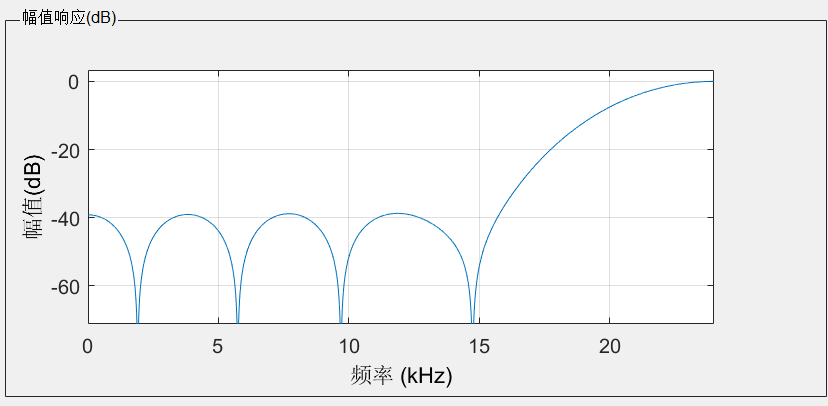
Since M = N – 1, which M equals to order minus 1. We get M equal to 4.



This is Type II

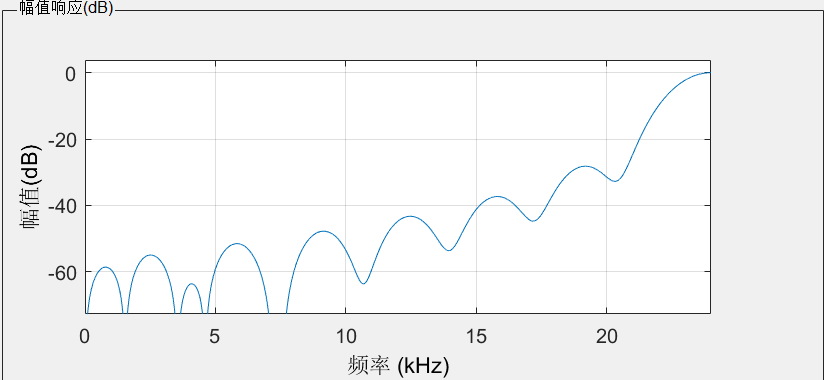
1.2.2（low pass filter will be better）

Keeping using the hamming window:



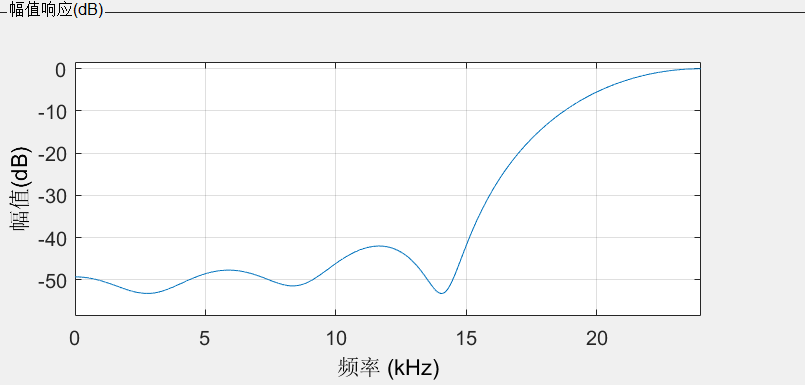
In this magnitude responses, order is 12 and Fc is 23.

Using the Bartlett window we can get:



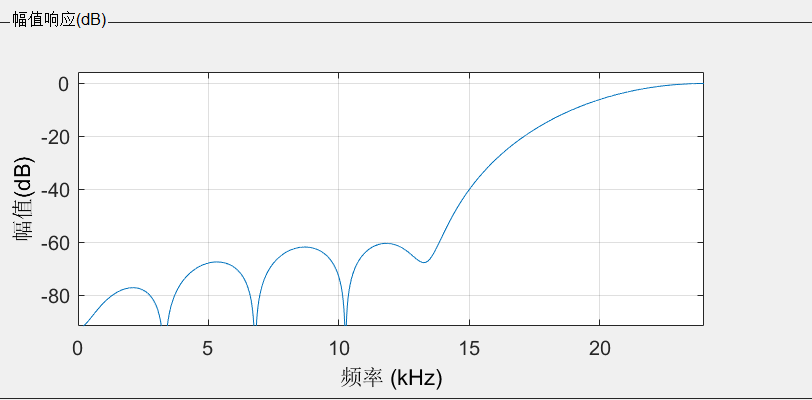
In this magnitude responses, order is 29 and Fc is 23.

Using the Bartlett-Hanning window we can get:



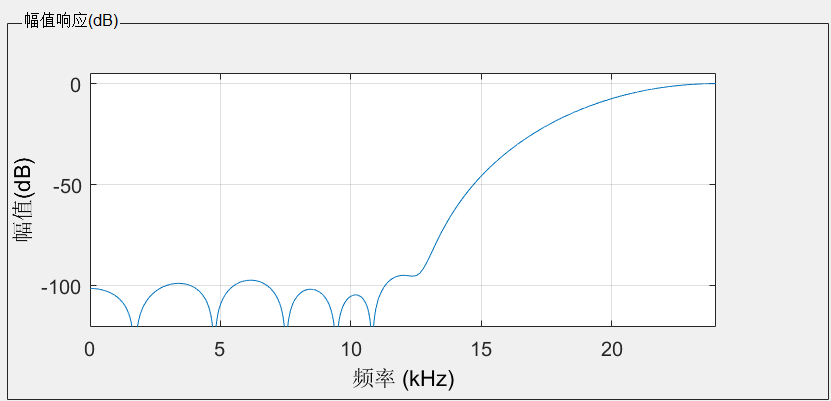
In this magnitude responses, order is 16 and Fc is 20.

Using the Blackman window we can get:



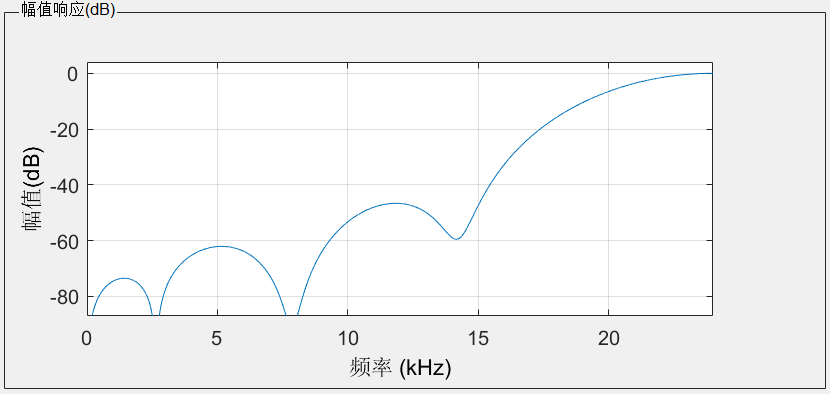
In this magnitude responses, order is 14 and Fc is 23.

Using the Blackman window we can get:



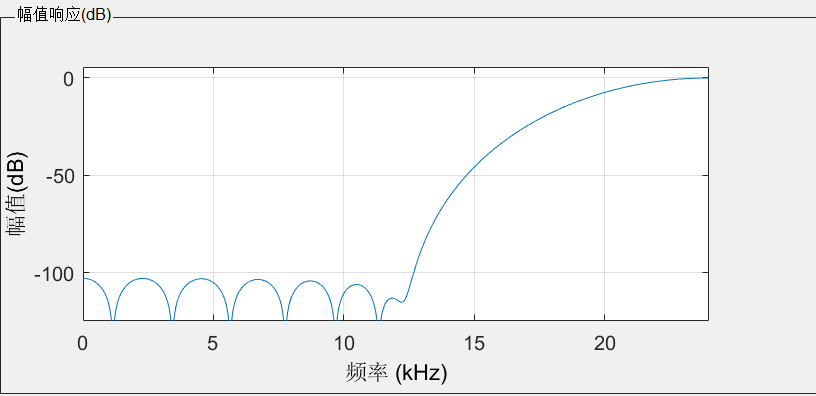
In this magnitude responses, order is 18 and Fc is 23.

Using the Bohman window we can get:



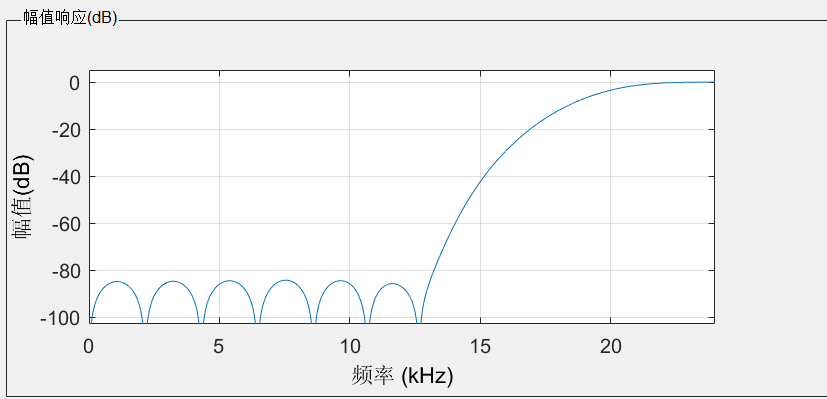
In this magnitude responses, order is 15 and Fc is 23.

Using the Chebyshev window we can get:



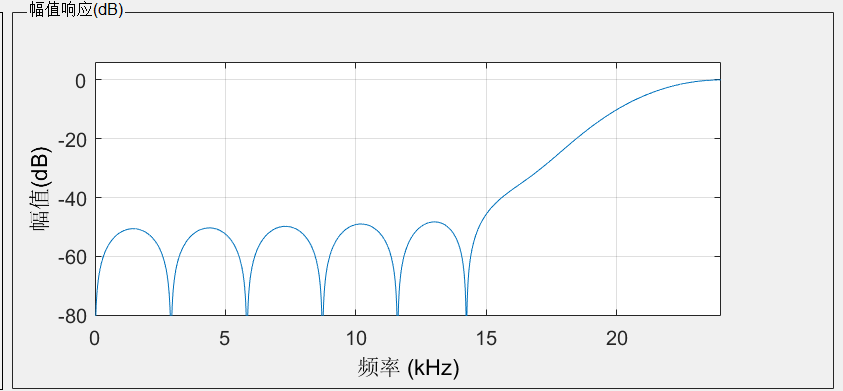
In this magnitude responses, order is 16 and Fc is 23.

Using the Flat-top window we can get:



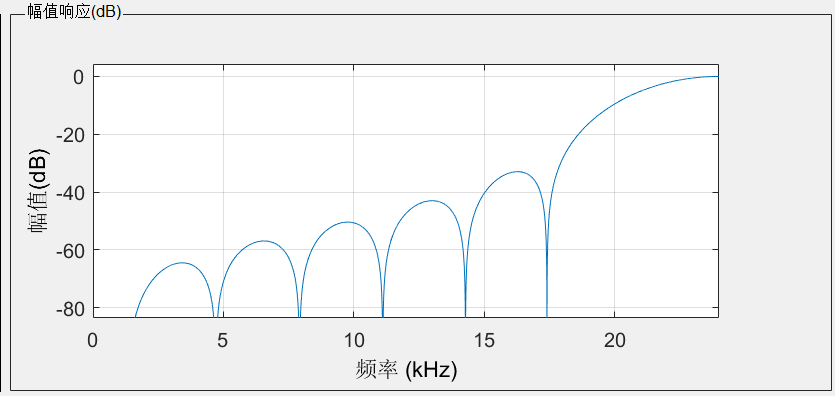
In this magnitude responses, order is 23 and Fc is 23.

Using the Gaussian window we can get:



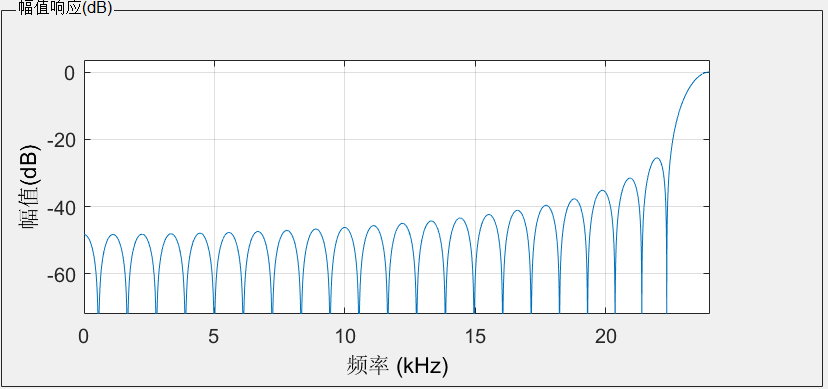
In this magnitude responses, order is 15 and Fc is 23.

Using the Hann window we can get:



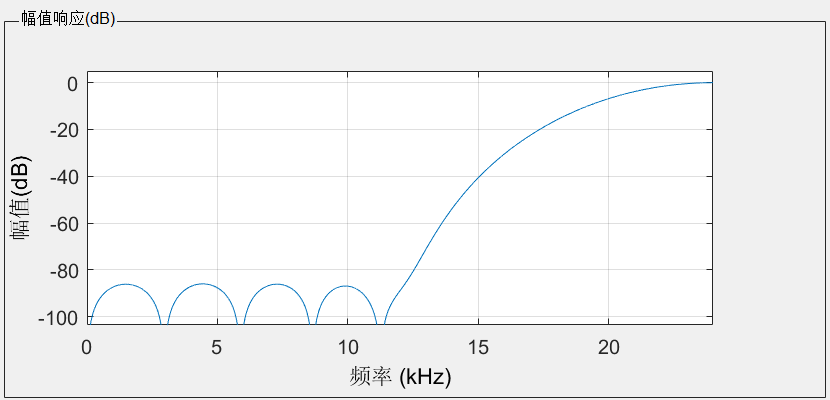
In this magnitude responses, order is 15 and Fc is 23.

Using the Kaiser window we can get:



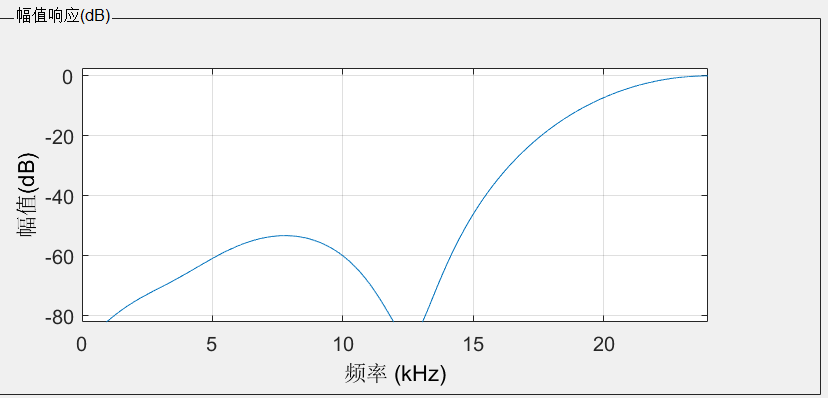
In this magnitude responses, order is 42 and Fc is 23.

Using the Nuttall window we can get:



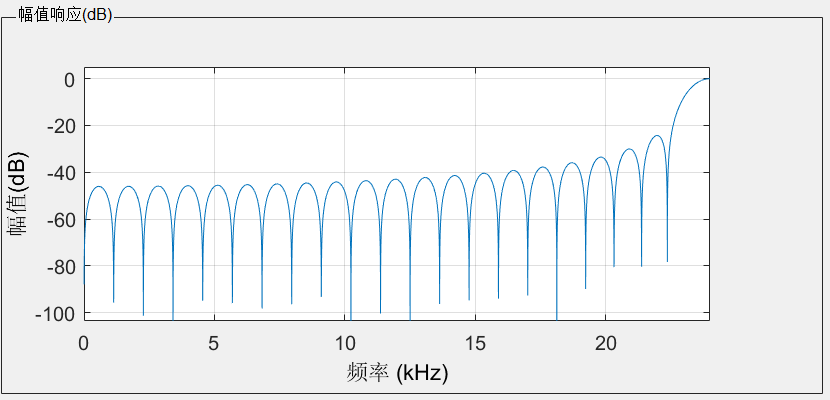
In this magnitude responses, order is 17 and Fc is 23.

Using the Parzen window we can get:



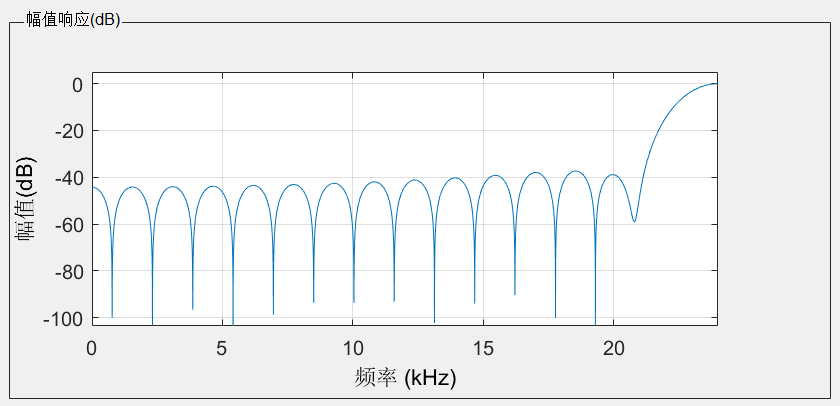
In this magnitude responses, order is 16 and Fc is 23.

Using the Rectangular window we can get:



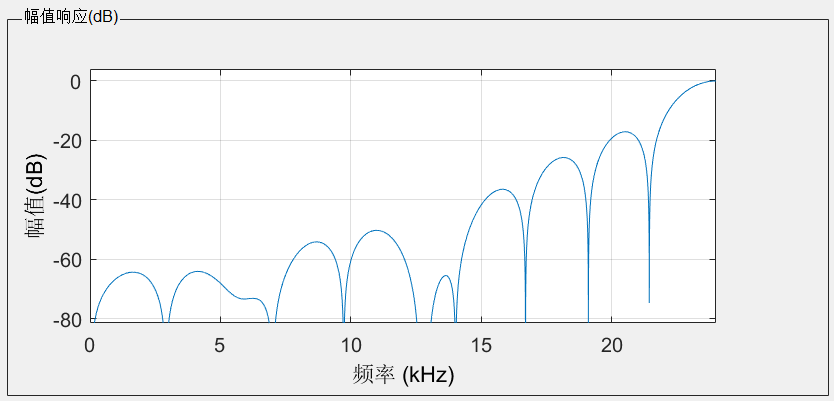
In this magnitude responses, order is 41 and Fc is 23.

Using the Taylor window we can get:



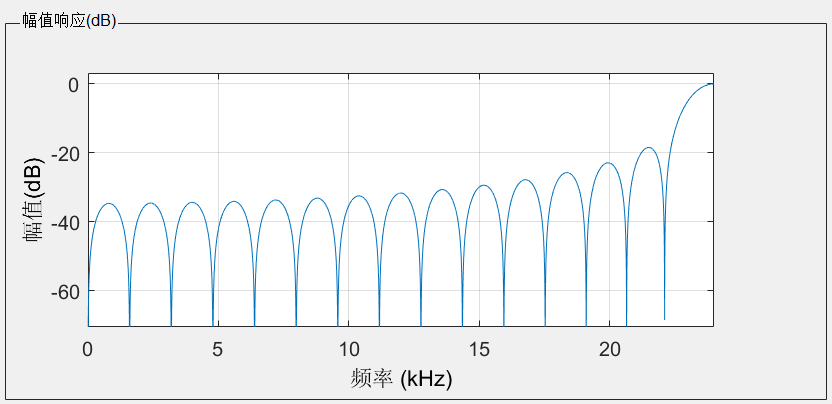
In this magnitude responses, order is 30 and Fc is 23.

Using the Tukey window we can get:



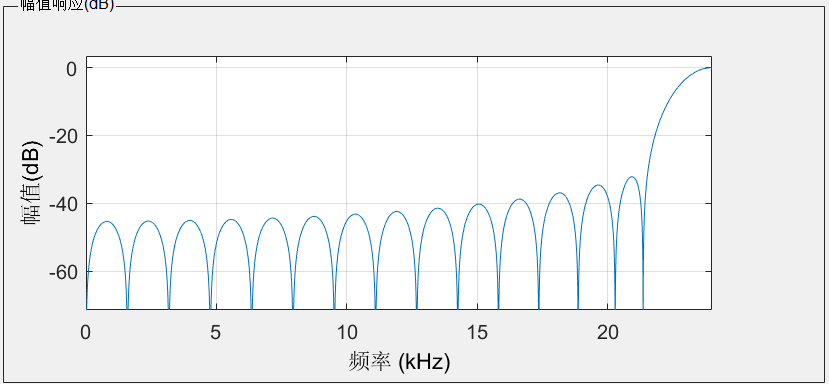
In this magnitude responses, order is 27 and Fc is 23.

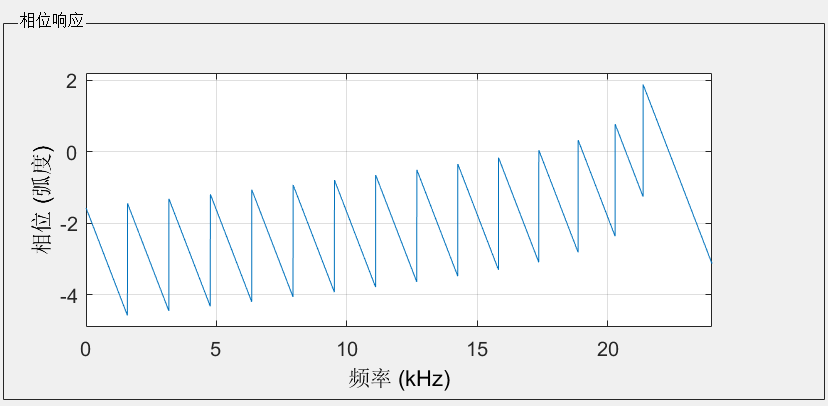
When choose the same order as Bartlett filter, using the Kaiser window we can get:



In this magnitude responses, order is 29 and Fc is 23 and beta is 0.5.

When the beta increases, the attenuation is increasing. When the beta is 2.8, the signal matches the role that signal has 40 dB attenuation before 15 kHz. The signal shows like:

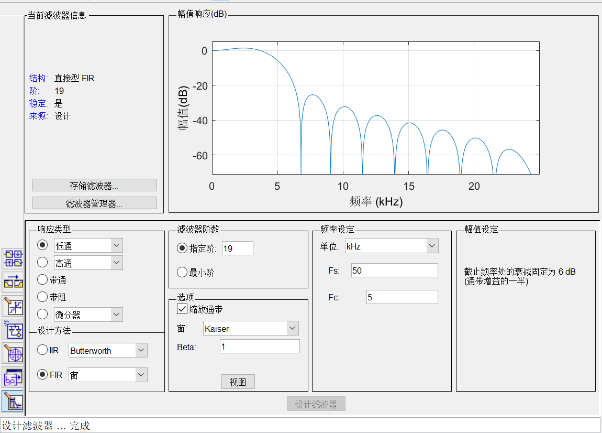




1.2.3

When we choose Fs = 50kHz, fc = 5kHz. From the given conditions we can have the equation:

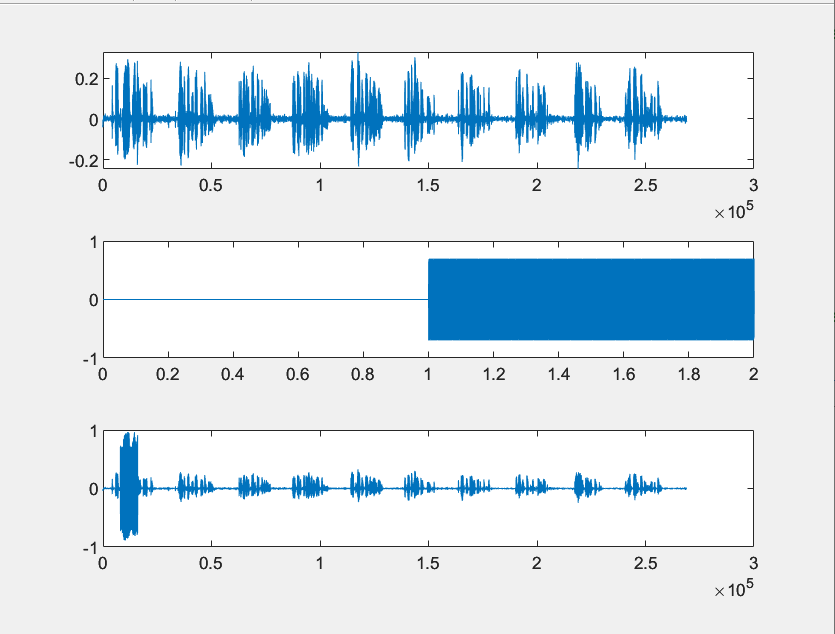
So when we choose order that near 18, the output will be better than other orders. The picture I take has the order 19 and It shows that:



From the exploration we can find that when using this equation to find parameters, It will help us find the rather satisfying parameters than we try various of parameters randomly.

1.3.1

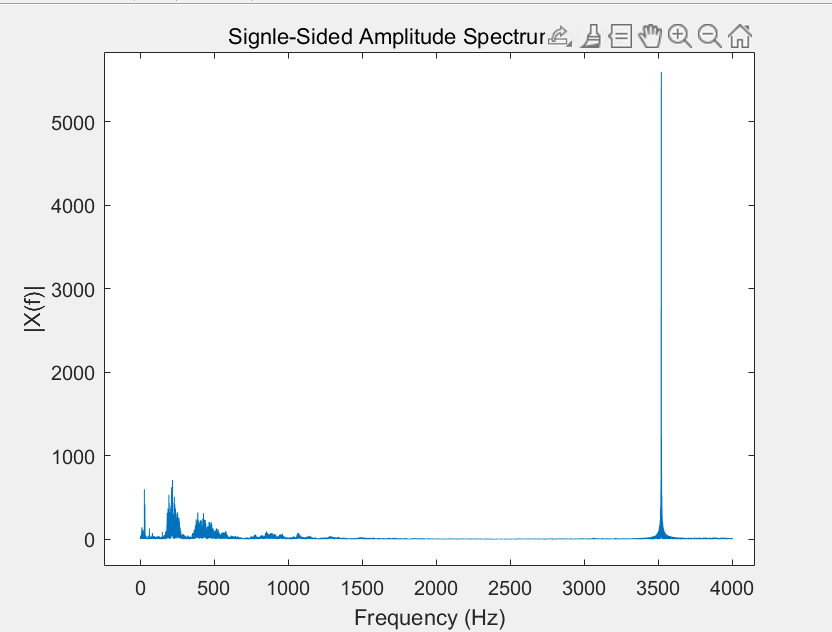
In this part we create a noise by createNote function. Then we add the noise in the speech so we can get a degraded signal. The final result is shows here:



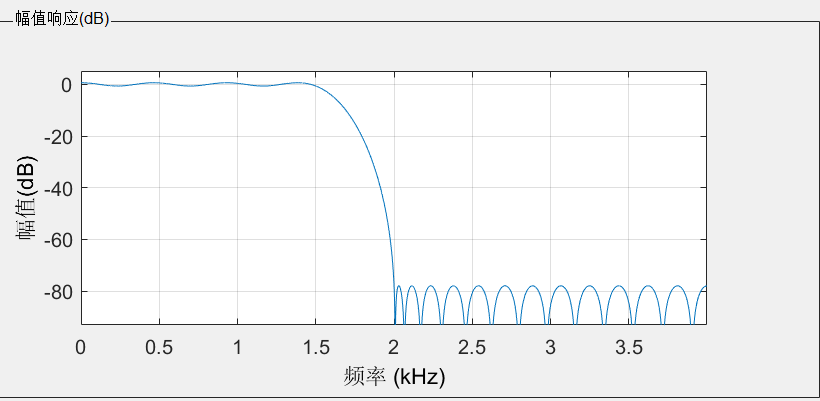
In this figure, (3,1,1) is the original signal, (3,1,2) is the noise signal created from the function createNote. (3,1,3) is the degraded signal.

1.3.2

In order to use the filter, we need to build the frquency figure first. So we build the figure like this:



So we can see that the noise that we need to clear shows approximately at 3500Hz. So I choose to build the low-pass filter:



Parameters I use:

Fs = 8000

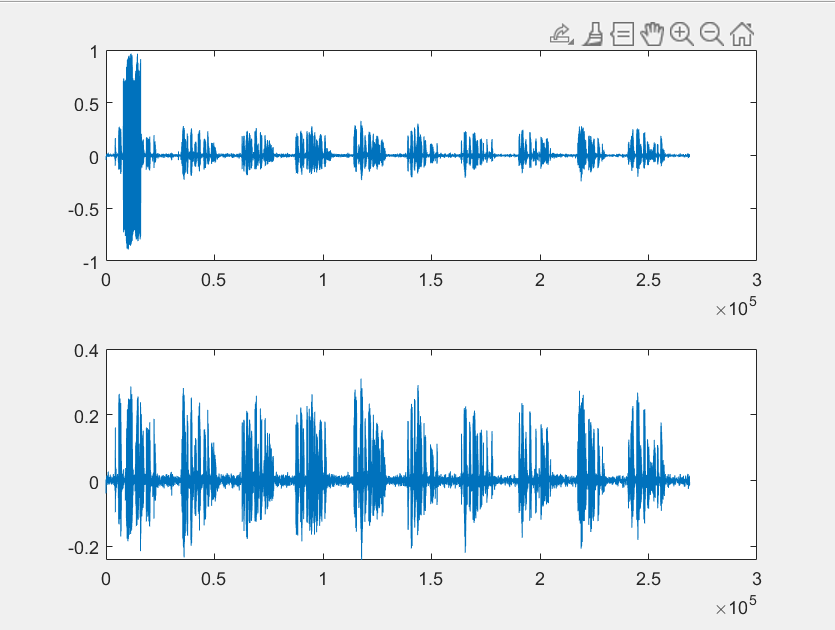
Fpass = 1500

Fstop = 2000

Apass = 1

Astop = 80

And we can get new voice:



We can see clearly that the noise is cleared completely.

For test the Fpass and Fstop, we change the Fpass and Fstop from 1000 to 3500.(since the noise happen over 3500Hz) We can see that although all the low-pass filter can clear the signal, but as Fpass becomes larger, the voice of the file becomes more clearly since we reduce the annuation to the original signal. But, at the same time, we also find that when we choose the bandwidth(absolute value of (Fpass – Fstop)) too narrow, the order of our filter will increase rapidly, which is really a bad circumstance to us. So after several detections, we choose to use parameters:

Fs = 8000

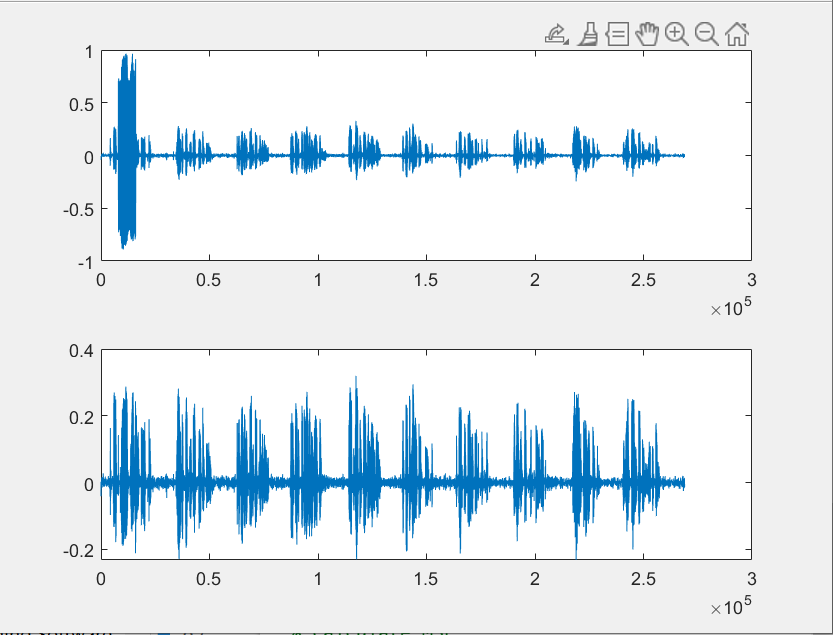
Fpass = 3000

Fstop = 3500

Apass = 1

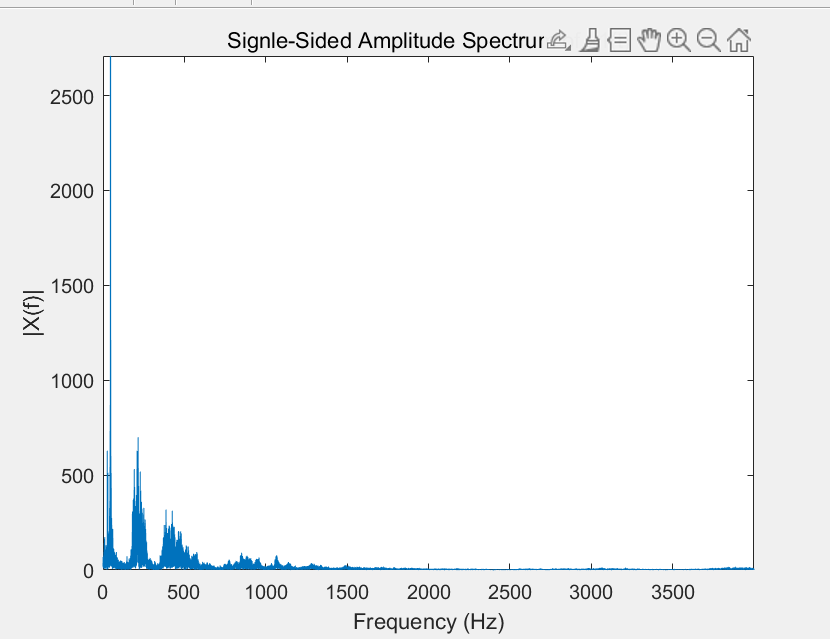
Astop = 80

With these parameters, the order of the filter is 40 and we can get the voice almost like the original wav.file. MSE is 1.9766e-12. Here is the picture that compares original signal and restored signal.



1.3.3

When we change the number of notes under 40( for me I use 30), the noise shows in a completely position at frequency domain:



So what we need to do is change the filter to a high-pass filter. After the same Inquiry Methodology as 1.3.2, we get parameters:

Fs = 8000

Fpass = 50

Fstop = 300

Apass = 1

Astop = 80

In this way we can get a clear and loud enough signal with noise disappeared. And MSE is 3.0429e-10. I think this meet the requirements that perfectly recover the original signal.

1.3.4

In this question, I use 1.3.2 and 1.3.3 to make super\_imposed tones. To clear the signal, I choose to filter them seperated and finally add them together. And it shows a better result than filter them in one time(which causes some remaining noise).