University of Electronic Science and Technology of China

Post graduate studies faculty

School of Resources and Environment

Filter specifications and how to design a digital filter based on the given specifications

Submitted To:

Assoc. Prof. Jing-Ran Lin（林静然）

Submitted By:

MERGHEM Faicel ID Number: 201224180104

- *Master student 2013*-

INTRODUCTION

This paper considers the problem of designing a digital filter. The design process begins with the filter specifications, which may include constraints on the magnitude and/or phase of the frequency response, constraints on the unit sample response or step response of the filter, specification of the type of filter (e.g., FIR or IIR), and the filter order. Once the specifications have been defined, the next step is to find a set of filter coefficients that produce an acceptable filter. After the filter has been designed, the last step is to implement the system in hardware or software, quantizing the filter coefficients if necessary, and choosing an appropriate filter structure

FILTER SPECIFICATIONS

Before a filter can be designed, a set of filter specifications must be defined. For example, suppose that we would like to design a low-pass filter with a cutoff frequencyω_c. The frequency response of an ideal low-pass filter with linear phase and a cutoff frequency ω_c is

$$H(e^{j\omega}) = \begin{cases} e^{-j\alpha\omega} & |\omega| < \omega_c \\ 0 & otherwise \end{cases}$$

Which has a unit sample response

$$h\ (n)= \frac{sin(n-\boldsymbol{a})\boldsymbol{w}}{\pi(n-\alpha)}$$

Because this filter is unrealizable (non-causal and unstable), it is necessary to relax the ideal constraints on the frequency response and allow some deviation from the ideal response. The specifications for a low-pass filter will illustrate in Fig. 1. Thus, the specifications include the pass band cutoff frequencyω_p, the stop band cutoff frequencyω_s, the pass band deviationδ_p and the stop band deviation δ_s , the passband and stopband deviations are often given in decibels (dB) as follows:

$$\propto_p = -20\log(1-\delta_p)$$

$$\propto_s = -20\log(\delta_s)$$

The interval between $[\omega_p, \omega_s]$ is called transition band

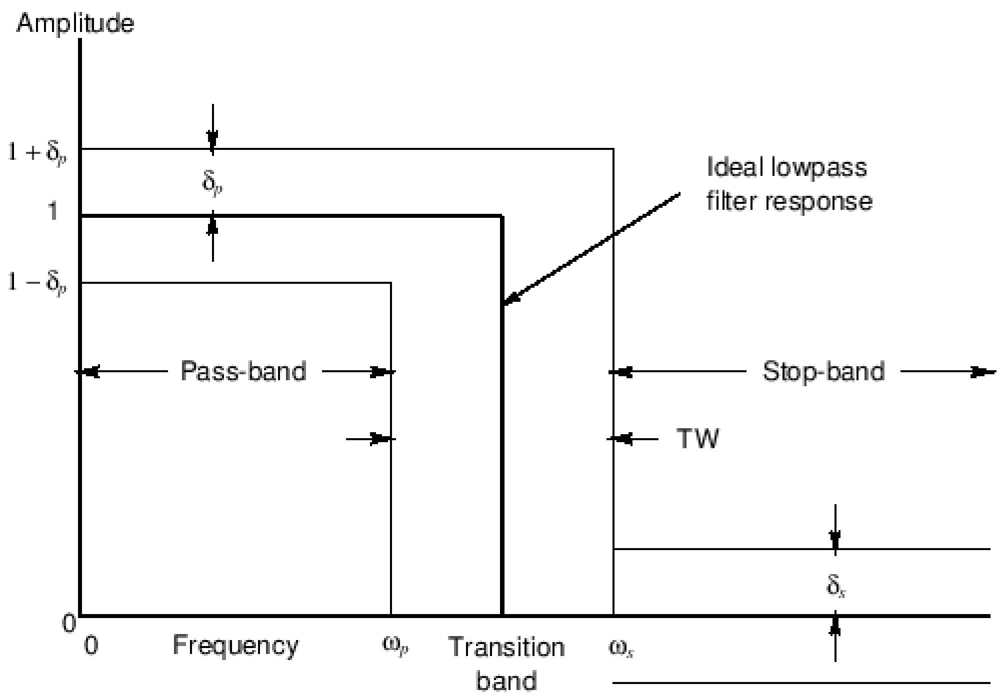


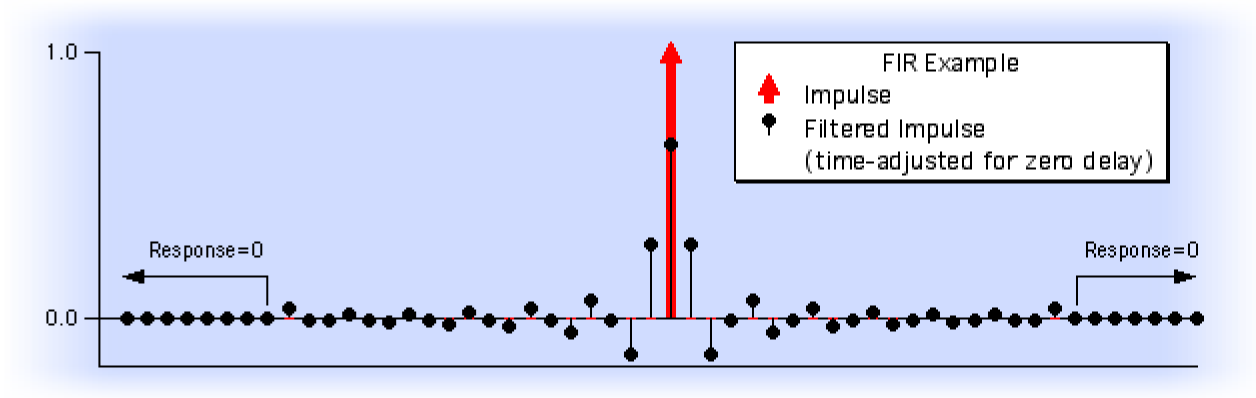
Figure 1: Illustration of typical ***low pass filter*** specifications in the ***frequency domain.***

How to design digital filter

Digital filters generally come in two flavors: Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters. Each one can implement a filter that passes or rejects bands of frequencies, but the mathematics and implementations differ significantly.

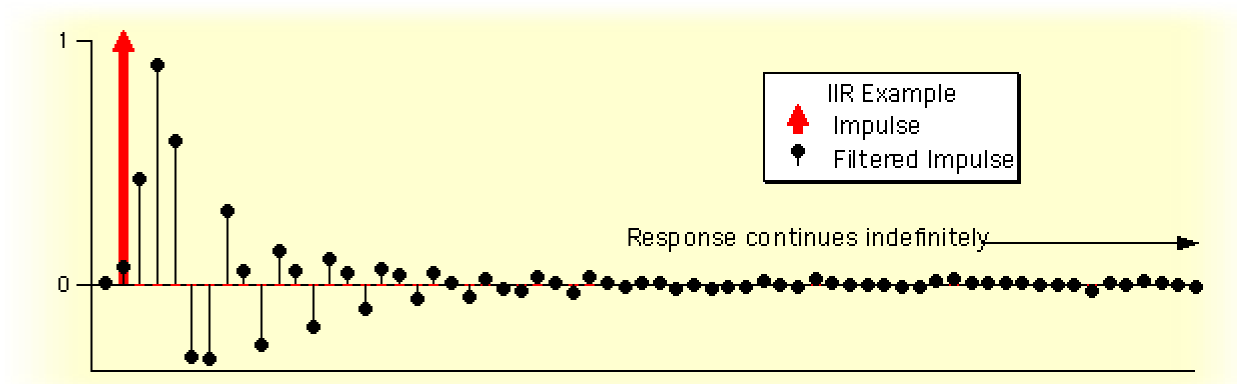
Finite Impulse Response Filters

"Finite Impulse Response" means that the filter´s time-domain response to an impulse (or "spike") is zero after a finite amount of time:



Infinite Impulse Response Filters

The response of an IIR filter continues indefinitely, as it does for analog electronic filters that employ inductors and capacitors:



To solve this topic question initially you should know how to select type of your filter, because of that, it's better to make comparison between IIR and FIR digital filter

FIR	***IIR***
– Always stable	– Possible to unstable
– Has a linear phase	– No linear phase
– High order	– Low order
– May be realized in both recursive	– Easily realized recursively

and non recursive structures. – No feedback	– Feedback

In many applications, the linearity of the phase response of digital filter is not an issue, making IIR filter preferable because of lower computational requirement. But in certain applications such as speech processing and data transmission, it's desirable to design linear phase.

The second step is estimate of the filter order; because of high order of FIR there are several method to estimate the lowest order of the filter. One of it is called Kaiser Formula

$$N = \frac{-20log_{10}(\sqrt{\delta_s \delta_p} - 13}{14.6(\omega_p - \omega_s)/2\pi}$$

From above equation the FIR filter is inversely proportional to the transition band, and directly with product$\delta_s \delta_p$, this illustrate the high order of transition band

Method to transform from H(s) to H(z):

1. *Back word integration*

   H(z)=$H(s)_{s=\frac{z-1}{Tz}}$

   T= sampling time

2. *Bilinear transformation integration*

The bilinear transformation is a mapping from the s-plane to the z-plane

H(z)=$H(s)_{s=\frac{2}{T}\frac{z-1}{Tz}}$

By this method we can transform of typical analog filter magnitude to digital filter magnitude response

Step of design using Bilinear transformation integration:

First you should pre warp the critical band edge frequencies and their analog equivalent to analog proto type $H_a(s)$

$$\Omega_p = \tan\frac{\omega_p}{2}$$

$$\Omega_s = \tan\frac{\omega_s}{2}$$

Second step is design analog proto type $H_a(s)$

The third step is transform $H_a(s)$ using Bilinear transformation integration to obtain G(z)

These three step are used to design low pass IIR digital filter but if you want to design HPF, BPF, and BSF IIR digital filters you should do one of two approaches.

- *The first approach consists of these steps:*
  1. Pre warp the digital frequency specification $G_D(z)$ to frequency specification of an analog type $H_D(s)$
  2. Convert the frequency specification $H_D(s)$ to analog proto type $H_{LP}(s)$
  3. Design $H_{LP}(s)$
  4. Convert $H_{LP}(s)$ to $H_D(s)$
  5. Transform $H_D(s)$ to $G_D(z)$
- *The second approach consists of this steps:*
  1. Pre warp the digital frequency specification $G_D(z)$ to frequency specification of an analog type $H_D(s)$
  2. Convert the frequency specification $H_D(s)$ to analog proto type $H_{LP}(s)$
  3. Design $H_{LP}(s)$
  4. Convert $H_{LP}(s)$ to $H_{LP}(s)$
  5. Transform $H_{LP}(s)$ to $G_D(z)$

There are useful transformations from LPF to HPF, or from LPF to BPF or from LPF to BSF. The table below shows the Spectral Transformation of LPF with cutoff frequencyw_c.

Type of Transformation	Transformation	Parameters
Lowpass	$z^{-1} \longrightarrow \frac{z^{-1}-\alpha}{1-\alpha z^{-1}}$	ω_c = cutoff frequency of new filter $\alpha = \frac{\sin\left[\left(\omega_c' - \omega_c\right)/2\right]}{\sin\left[\left(\omega_c' + \omega_c\right)/2\right]}$
Highpass	$z^{-1} \longrightarrow -\frac{z^{-1}+\alpha}{1+\alpha z^{-1}}$	ω_c = cutoff frequency of new filter $\alpha = -\frac{\cos\left[\left(\omega_c' + \omega_c\right)/2\right]}{\cos\left[\left(\omega_c' - \omega_c\right)/2\right]}$
Bandpass	$z^{-1} \longrightarrow -\frac{z^{-2}-\alpha_1 z^{-1}+\alpha_2}{\alpha_2 z^{-2}-\alpha_1 z^{-1}+1}$	ω_ℓ = lower cutoff frequency ω_u = upper cutoff frequency $\alpha_1 = -2\beta K/(K+1)$ $\alpha_2 = (K-1)/(K+1)$ $\beta = \frac{\cos\left[\left(\omega_u + \omega_\ell\right)/2\right]}{\cos\left[\left(\omega_u - \omega_\ell\right)/2\right]}$ $K = \cot\frac{\omega_u - \omega_\ell}{2}\tan\frac{\omega_c'}{2}$
Bandstop	$z^{-1} \longrightarrow \frac{z^{-2}-\alpha_1 z^{-1}+\alpha_2}{\alpha_2 z^{-2}-\alpha_1 z^{-1}+1}$	ω_ℓ = lower cutoff frequency ω_u = upper cutoff frequency $\alpha_1 = -2\beta/(K+1)$ $\alpha_2 = (K-1)/(K+1)$ $\beta = \frac{\cos\left[\left(\omega_u + \omega_\ell\right)/2\right]}{\cos\left[\left(\omega_u - \omega_\ell\right)/2\right]}$ $K = \tan\frac{\omega_u - \omega_\ell}{2}\tan\frac{\omega_c'}{2}$

Table 1. *Spectral Transformation of LPF with cutoff frequency*w_c

Ideal filter approximation

The ideal filter frequency response is used when designing FIR filters using window functions. The objective is to compute the ideal filter samples. FIR filters have finite impulse response, which means the ideal filter frequency sampling must be performed in a finite number of points. As the ideal filter frequency response is infinite, it is easy to produce sampling errors. The error is less as the filter order increases.

Figure (2) illustrates the transfer functions of four standard ideal filters.

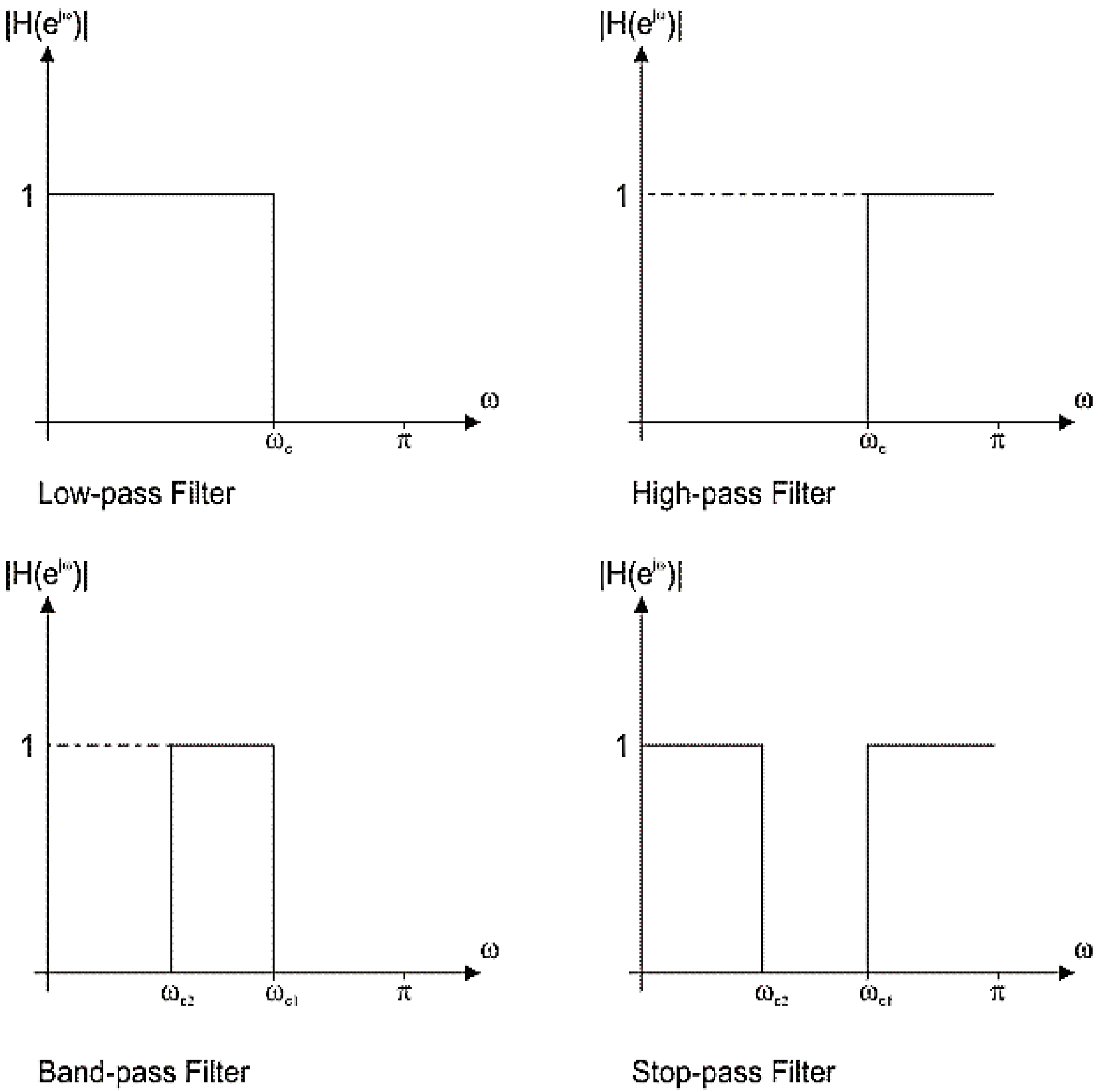


Figure 2. *Transfer functions of four standard ideal filters*

The ideal filter frequency response can be computed via inverse Fourier transform. The four standard ideal filters frequency responses are contained in the table 2 below.

Type of filter	Frequency response $h_d[n]$
low-pass filter	$h_d[n] = \begin{cases} \dfrac{\sin[\omega_c(n-M)]}{\pi(n-M)}; & n \neq M \\ \dfrac{\omega_c}{\pi}; & n = M \end{cases}$
high-pass filter	$h_d[n] = \begin{cases} 1-\dfrac{\omega_c}{\pi}; & n \neq M \\ -\dfrac{\sin(\omega_c(n-M))}{\pi(n-M)}; & n = M \end{cases}$
band-pass filter	$h_d[n] = \begin{cases} \dfrac{\sin(\omega_{c2}(n-M))}{\pi(n-M)} - \dfrac{\sin(\omega_{c1}(n-M))}{\pi(n-M)}; & n \neq M \\ \dfrac{\omega_{c2}-\omega_{c1}}{\pi}; & n = M \end{cases}$
band-stop filter	$h_d[n] = \begin{cases} \dfrac{\sin(\omega_{c1}(n-M))}{\pi(n-M)} - \dfrac{\sin(\omega_{c2}(n-M))}{\pi(n-M)}; & n \neq M \\ 1-\dfrac{\omega_{c2}-\omega_{c1}}{\pi}; & n = M \end{cases}$

Table 2. *The frequency responses of four standard ideal filters*

The value of variable **n** ranges between 0 and N, where N is the filter order. A constant M can be expressed as M = N / 2. Equivalently, N can be expressed as N = 2M.

The constant M is an integer if the filter order N is even, which is not the case with odd order filters. If M is an integer (even filter order), the ideal filter frequency response is symmetric about its Mth sample which is found via expression shown in the table 2-2-1 above. If M is not an integer, the ideal filter frequency response is still symmetric, but not about some frequency response sample.

Since the variable **n** ranges between 0 and N, the ideal filter frequency response has N+1 sample.

If it is needed to find frequency response of a non-standard ideal filter, the expression for inverse Fourier transform must be used:

$$h_d[n] = \frac{1}{\pi}\int_0^{\pi} e^{j\omega(n-M)}d\omega$$

Non-standard filters are rarely used. However, if there is a need to use some of them, the integral above must be computed via various numerical methods.

FIR filter design using window functions

The FIR filter design process via window functions can be split into several steps:

1. Defining filter specifications.
2. Specifying a window function according to the filter specifications.
3. Computing the filter order required for a given set of specifications.
4. Computing the window function coefficients.
5. Computing the ideal filter coefficients according to the filter order.
6. Computing FIR filter coefficients according to the obtained window function and ideal filter coefficients.
7. If the resulting filter has too wide or too narrow transition region, it is necessary to change the filter order by increasing or decreasing it according to needs, and after that steps 4, 5 and 6 are iterated as many times as needed.

The final objective of defining filter specifications is to find the desired normalized frequencies (ω_c, ω_{c1}, ω_{c2}), transition width and stop band attenuation. The window function and filter order are both specified according to these parameters.

Accordingly, the selected window function must satisfy the given specifications. After this step, that is, when the window function is known, we can compute the filter order required for a given set of specifications. When both the window function and filter order are known, it is possible to calculate the window function coefficients w[n] using the formula for the specified window function.

After estimating the window function coefficients, it is necessary to find the ideal filter frequency samples. The final objective of this step is to obtain the coefficients $h_d(n)$. Two sequencies w[n] and $h_d(n)$ have the same number of elements.

The next step is to compute the frequency response of designed filter h[n] using the following expression:

$$h[n] = w[n] \cdot h_d[n]$$

Lastly, the transfer function of designed filter will be found by transforming impulse response via Fourier transform:

$$H(e^{j\omega}) = \sum_{n=0}^{N} h[n] \cdot e^{-jn\omega}$$

Or via Z-transform:

$$H(z) = \sum_{n=0}^{N} h[n] z^{-n}$$

If the transition region of designed filter is wider than needed, it is necessary to increase the filter order, reestimate the window function coefficients and ideal filter frequency samples, multiply them in order to obtain the frequency response of designed filter and reestimate the transfer function as well. If the transition region is narrower than needed, the filter order can be decreased for the purpose of optimizing hardware and/or software resources. It is also necessary to reestimate the filter frequency coefficients after that. For the sake of precise estimates, the filter order should be decreased or increased by 1.

Properties of some fixed window functions:

Type of windows	Main lobe width ΔML	Relative sidelobe level Asl	Minimum stopband attenuation	Transition bandwidth ΔM
Rectangular	4π/(2M+1)	13.3dB	20.9dB	0.92 π/M
Hann	8π/(2M+1)	31.5dB	43.9dB	3.11 π/M
Hamming	8π/(2M+1)	42.7dB	54.5dB	3.32 π/M
Blackman	12π/(2M+1)	75.3dB	75.3dB	5.56 π/M

Conclusion

Digital filter is a system that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal.

Digital filter system usually consists of an analog-to-digital converter to sample the input signal, followed by a microprocessor and some peripheral components such as memory to store data and filter coefficients etc. Finally a digital-to-analog converter to complete the output stage. Program Instructions (software) running on the microprocessor implement the digital filter by performing the necessary mathematical operations on the numbers received from the ADC. In some high performance applications, an FPGA or ASIC is used instead of a general purpose microprocessor, or a specialized DSP with specific paralleled architecture for expediting operations such as filtering.

Digital filters may be more expensive than an equivalent analog filter due to their increased complexity, but they make practical many designs that are impractical or impossible as analog filters. Since digital filters use a sampling process and discrete-time processing, they experience latency (the difference in time between the input and the response), which is almost irrelevant in analog filters.