NAME: ABOAGYE ISAAC ADJAYE

(PhD. CONTINUE STUDENT)

STUDENT ID: 20I414010104

LECTURER: ASSOC. PROF. JING-RAN LIN

COURSE: DIGITAL SIGNAL PROCESSING

COUNTRY: GHANA

DEPARTMENT: COMMUNICATION ENGINEERING

E-MAIL: IKECRUSTY@YAHOO.COM

TOPIC:

FILTER SPECIFICATIONS AND HOW TO DESIGN A DIGITAL FILTER BASED ON THE GIVEN SPECIFICATIONS.

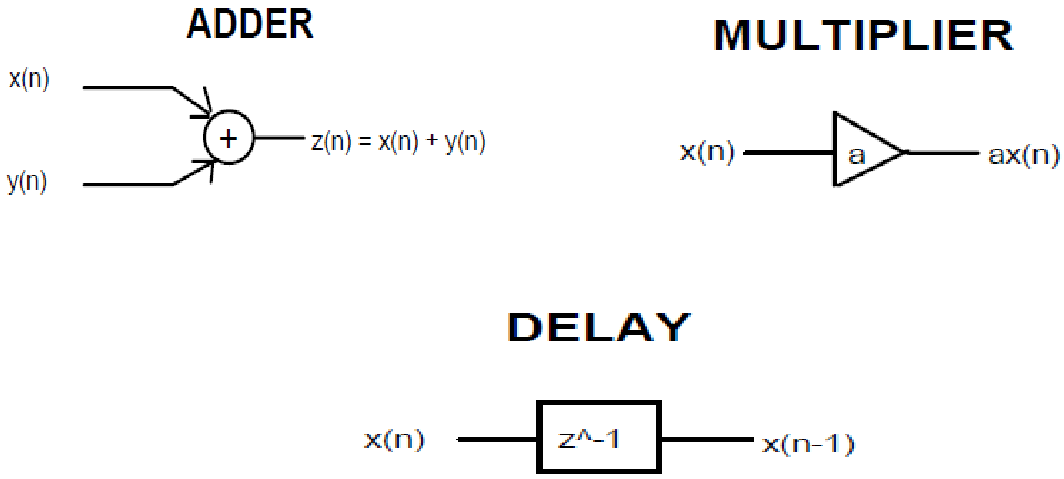
ABSTRACT

Filters can be built in a number of different technologies. The same transfer function can be realized in several different ways, that is the mathematical properties of the filter are the same but the physical properties are quite different. Often the components in different technologies are directly analogous to each other and fulfill the same role in their respective filters. Filtering operations include Noise suppression, Enhancement of selected frequency ranges, Band limiting etc. Finite impulse response filters are highly desirable in digital filter design because of their inherent stability and linear phase. However, when narrow transition band characteristics are required, they typically have a much higher filter order than their infinite impulse response counterparts with equivalent magnitude spectrums. This paper will investigate the specifications and the method to design a digital filter based on the given specifications**.** An important step in the development of a digital filter is the determination of a realizable transfer function G(z) approximating the given frequency response specifications. Infinite impulse response (IIR) digital filter design will be of particular interest since they typically meet a given set of specifications with a much lower filter order than a corresponding FIR filter. Moreover, order of an FIR filter in most cases is considerably higher than the order of an equivalent IIR filter meeting the same specifications, and FIR filter has thus higher computational complexity.

INTRODUCTION

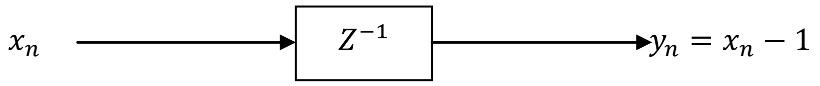
Digital Filter is a numerical procedure or algorithm that transforms a given sequence of numbers into a second sequence that has some more desirable properties. In signal processing, a filter is a device or process that removes from a signal some unwanted component or feature. Filtering is a class of signal processing, the defining feature of filters being the complete or partial suppression of some aspect of the signal. Most often, this means removing some frequencies and not others in order to suppress interfering signals and reduce background noise. However, filters do not exclusively act in the frequency domain; especially in the field of image processing, many other targets for filtering exist. The drawback of filtering is the loss of information associated with it. Signal combination in Fourier space is an alternative approach for removal of certain frequencies from the recorded signal. Filters may be: Analog or Digital, Discrete-time (sampled) or Continuous-time, Linear or Non-linear, Time-invariant or Time-variant, also known as shift invariance, Passive or Active type of continuous-time filter, Infinite impulse response (IIR) or Finite impulse response (FIR) type of discrete-time or digital filter . A key element in processing digital signals is the filter. Filters perform direct manipulations on the spectra of signals. To completely describe digital filters, three basic elements (or building blocks) are needed: **an Adder, a Multiplier, and a Delay element**. The adder has two inputs and one output, and it simply adds the two inputs together. The multiplier is a gain element, and it multiplies the input signal by a constant. The delay element delays the incoming signal by one sample. Digital filters can be implemented using either a block diagram or a signal flow graph. With the basic building blocks at hand, the two different filter structures can easily be implemented based on given specifications. These two structures are Infinite Impulse Response (IIR) and Finite Impulse Response (FIR), depending on the form of the system's response to a unit pulse input. IIR filters are commonly implemented using a feedback (recursive) structure, while FIR filters usually require no feedback (non-recursive). An important step in the development of a digital filter is the determination of a realizable transfer function G(z) approximating the given frequency response specifications. If an IIR filter is desired, it is also necessary to ensure that G(z) is stable. The process of delivering the transfer function G(z) is called digital filter design. After G(z) has been obtained, the next step is to realize it in the form of suitable filter structure. Two major issues need to be answered before one can develop the digital transfer function G(z). These are the development of a reasonable filter frequency response specification from the requirement of the overall system in which the digital filter is to be employed and to determine whether FIR or IIR digital filter is to be determined. In the design of IIR filters, a commonly used approach is called the **bilinear transformation**. This design begins with the transfer function of an analog filter, and then performs a mapping from the s-domain to the z-domain. There is an alternative method for designing IIR digital filters that is **spectral transformation**. It replaces the analog frequency transformation by a frequency transformation in the digital domain. The basic elements of a digital filter are shown below:

ELEMENTS OF A DIGITAL FILTER



Positive delay ("delay"):

Stores the current value for one sample interval



Negative delay ("advance"):

Allows to look ahead, e.g. image processing

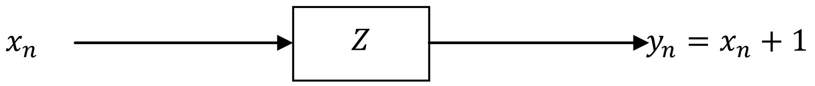


Fig. 1: Block Diagram of Filter Elements.

Signal Flow Graph of Filter Elements

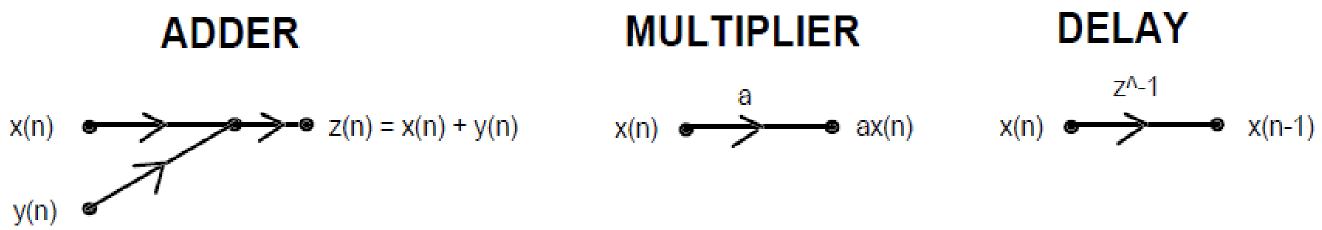


Fig. 2: Signal Flow Graph of Filter Elements.

DIGITAL FILTER DESIGN CONSIDERATIONS

Digital filter design is the process of deriving the transfer function $G(z)$, usually, either the magnitude and/or the phase (delay) response is specified for the design of digital filter for most applications. The basic considerations involved in digital filter design include the following:

1. Specification of the filter's response,
2. Design of transfer function of the filter,
3. Verification of the filter's performance by, analytic means, simulations and testing with real data if possible,
4. Implementation by hardware / software (or both).

DIGITAL FILTER SPECIFICATIONS

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

Table 1. The Frequency Responses of Four Standard Ideal Filters

Type of Filter	**Frequency Response** $h_d[h]$
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	$h_d[h]=\begin{cases}1-\dfrac{\omega_c}{\pi}; & n \neq M \\ \dfrac{-\sin(\omega_c(n-M))}{\pi(n-M)}; & n = M\end{cases}$
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	$h_d[n]=\begin{cases}\dfrac{\sin(\omega_{c1}(n-M))}{\pi(n-M)}-\dfrac{\sin(\omega_{c2}(n-M))}{\pi(n-M)}; & n \neq M \\ 1-\dfrac{\omega_{c2}-\omega_{c1}}{\pi}; & n = M\end{cases}$

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. 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:

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

In most practical applications, the problem of interest is the development of a realizable approximation to a given magnitude response specification. Basically, there are four types of ideal filters with magnitude response as shown below:

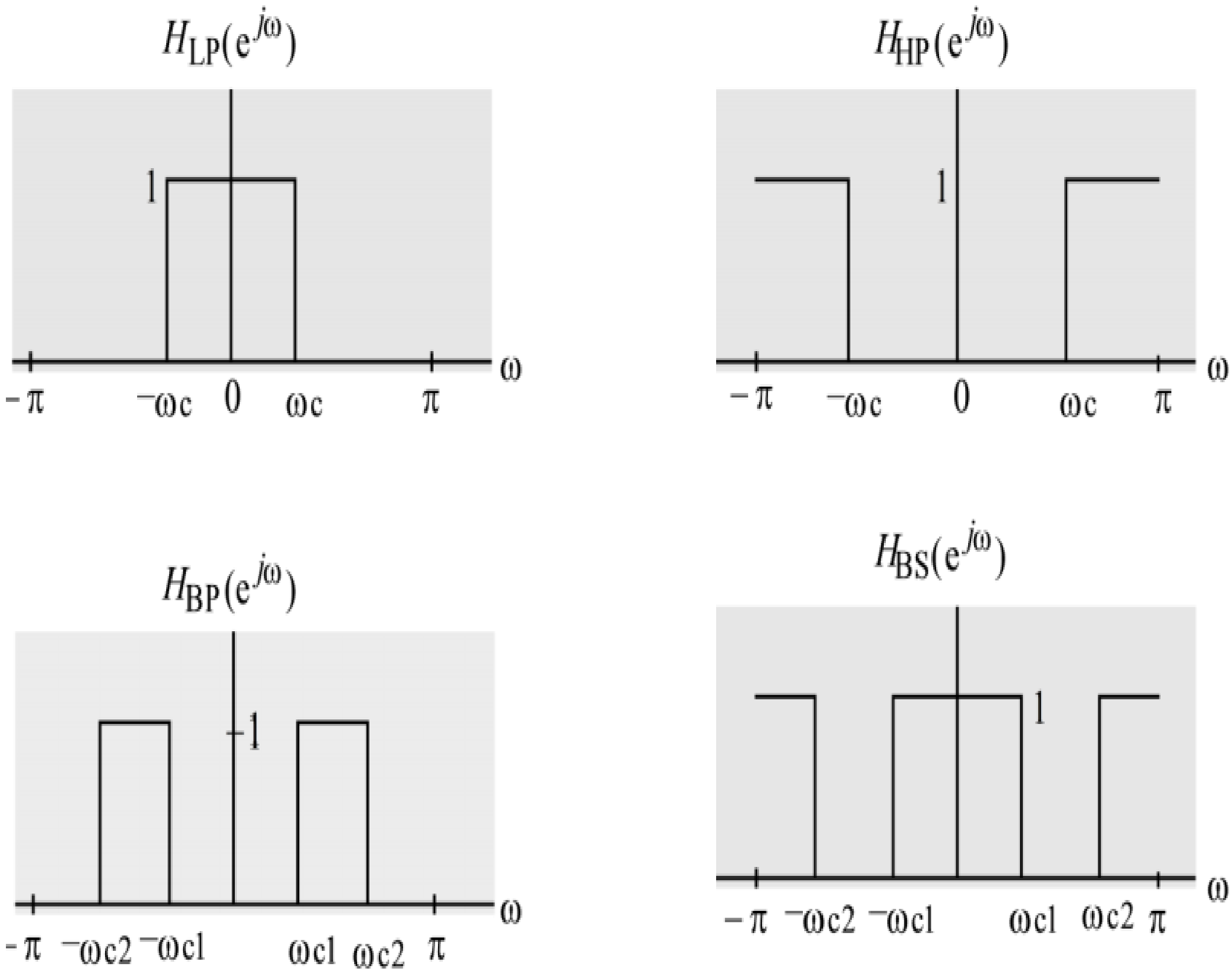


Fig. 3: Types of Ideal Filters with Magnitude Response.

As the impulse response corresponding to each of these ideal filters is noncausal and of infinite length, these filters are not realizable. In practice, the magnitude response specifications of a digital filter in the passband and in the stopband are given with some acceptable tolerances. In addition, a transition band is specified between the passband and stopband. Another perspective that provides some understanding can be obtained by looking at the ideal amplitude squared. When designing a filter, there is always an end goal in mind that has certain specifications. There are several methods for determining the minimum length a filter needs to be to meet these given specifications. However, these are called estimations for a reason because they do not always provide the correct filter order. The smallest integer value that lies above the estimation should be checked for accuracy after the implementation. For example, the magnitude response $\left|G\left(e^{j\omega}\right)\right|$ of a digital lowpass filter may be given as in figure 4 below. As indicated in the figure, in the passband, defined by $0\leq\omega\leq\omega_p$, we require that $\left|G\left(e^{j\omega}\right)\right| \cong 1$ with an error $\pm\delta_p$, i.e. $1-\delta_p \leq \left|G\left(e^{j\omega}\right)\right| \leq 1+\delta_p, |\omega| \leq \omega_p$.

In the stopband, defined by $\omega_s \leq\omega\leq\pi$, we require that $\left|G\left(e^{j\omega}\right)\right| \cong 0$ with an error δ_s

i.e., $\left|G\left(e^{j\omega}\right)\right| \leq \delta_s, \quad \omega \leq |\omega| \leq \pi$.

The parameters given include normalized passband edge angular frequency ω_p, normalized stopband edge angular frequency ω_s, peak passband ripple δ_p, and peak stopband ripple δs.

Now, the frequency response $\left|G\left(e^{j\omega}\right)\right|$ is a periodic function of ω and the magnitude response of a real-coefficient digital filter is an even function of ω_p,. As a result, filter specifications are given only for the frequency range $0 \leq |\omega| \leq \pi$.

Often, digital filter specifications are given in terms of loss function $G(\omega) = -20\log_{10}\left|G\left(e^{j\omega}\right)\right|$ in dB Peak passband ripple $\alpha_p = -20\log_{10}\left(1-\delta_p\right)$dB.

Minimum stopband attenuation $\alpha_s = -20\log_{10}\left(\delta_s\right)$dB

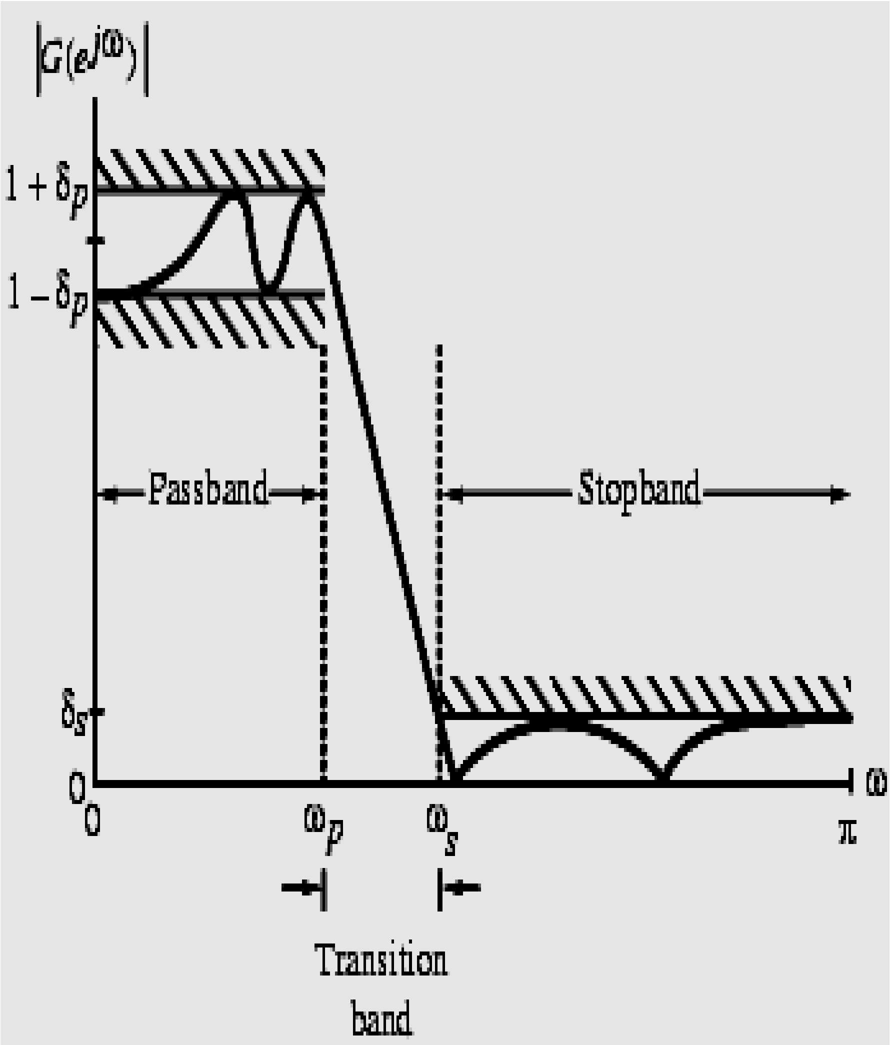


Fig. 4: Magnitude Specifications for a Digital Lowpass Filter.

Now, the specifications for a digital lowpass filter may alternatively be given in terms of its magnitude response, as in figure 5. Here, the maximum value of the magnitude in the passband is assumed to be unity, and the maximum passband deviation, denoted as $1/\sqrt{(1+\varepsilon^2)}$, is given by the minimum value of the magnitude in the passband. The maximum stopband is denoted by $1/A$. For the normalized specification, the maximum value of the gain function or the minimum value of the loss function is therefore 0dB. The quantity α_{max} given $\alpha_{max} = 20\log_{10}\left(\sqrt{1+\varepsilon^2}\right)$dB is called the minimum passband attenuation. For $\delta_p << 1$, it can be shown that $\alpha_{max} \cong -20\log_{10}(1-2\delta_p)$dB

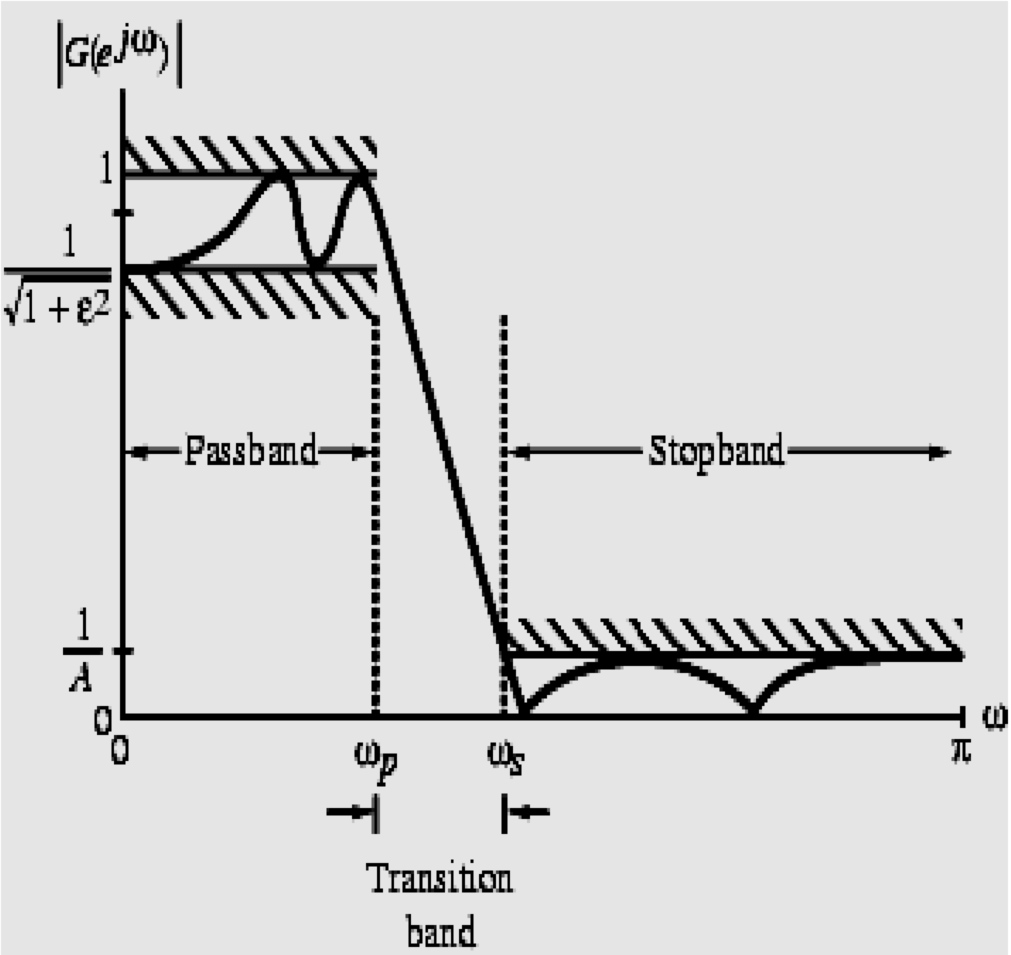


Fig. 5: Alternate magnitude specifications for a digital lowpass filter.

The passband and stopband edge frequencies, in most applications, are specified in Hz, along with the sampling rate of the digital filter. Since all filter design techniques are developed in terms of the angular frequencies ω_P and ω_S, the specified critical frequencies need to be normalized before a specific filter design algorithm can be applied. In practice, passband edge frequency F_P and stopband edge frequency F_s are specified in Hz. For digital filter design, normalized band edge frequencies need to be computed from specifications in Hz. Using:

$$\omega_P = \frac{\Omega_P}{F_T} = \frac{2\pi F_p}{F_T} = 2\pi F_P T$$

$$\omega_s = \frac{\Omega_P}{F_T} = \frac{2\pi F_s}{F_T} = 2\pi F_s T$$

BASIC CONCEPTS AND FILTER SPECIFICATION

Figure 6 illustrates a low-pass digital filter specification. The word specification actually refers to the frequency response specification.

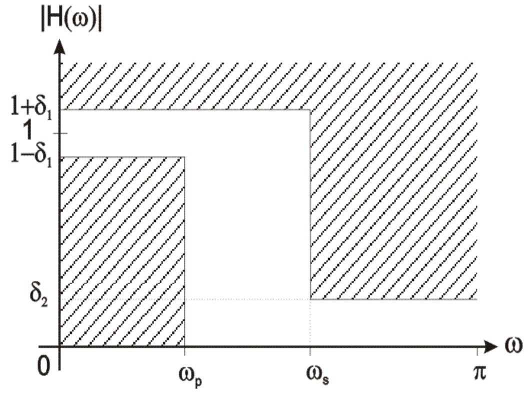


Fig. 6a. Low-pass digital filter specification.

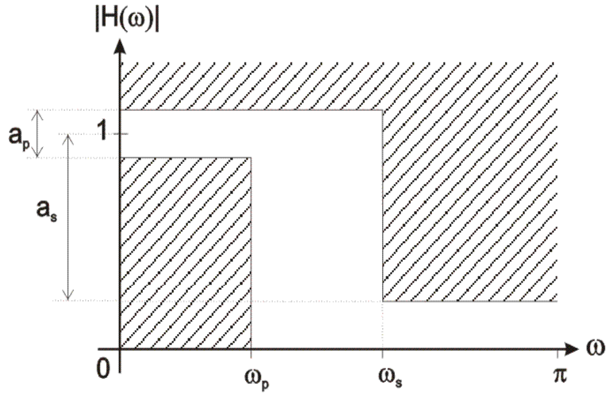


Fig. 6b. Low-pass digital filter specification.

- ωp – normalized cut-off frequency in the passband;
- ωs – normalized cut-off frequency in the stopband;
- $\delta 1$ – maximum ripples in the passband;
- $\delta 2$ – minimum attenuation in the stopband $[\mathrm{dB}]$;
- ap – maximum ripples in the passband; and
- as – minimum attenuation in the stopband $[\mathrm{dB}]$.

$$a_p = 20\log_{10}\left(\frac{1+\delta_1}{1-\delta_1}\right)$$

$$a_s = -20\log_{10}\delta_2$$

Frequency normalization can be expressed as follows:

$$\omega = \frac{2\pi f}{f_s}$$

where:

- fs is a sampling frequency;
- f is a frequency to normalize; and

- ω is normalized frequency.

Specifications for high-pass, band-pass and band-stop filters are defined almost the same way as those for low-pass filters. Figure 7 illustrates a high-pass filter specification.

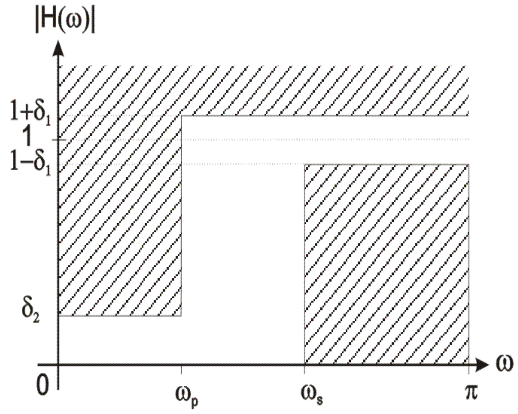


Fig. 7a. High-pass digital filter specification.

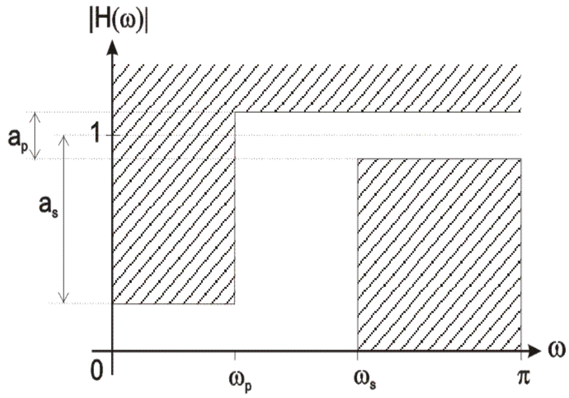


Fig.7b. High-pass digital filter specification.

Comparing these two figures 6 and 7, it is obvious that low-pass and high-pass filters have similar specifications. The same values are defined in both cases with the difference that in the later case the passband is substituted by the stopband and vice versa. Figure 8 illustrates a band-pass filter specification.

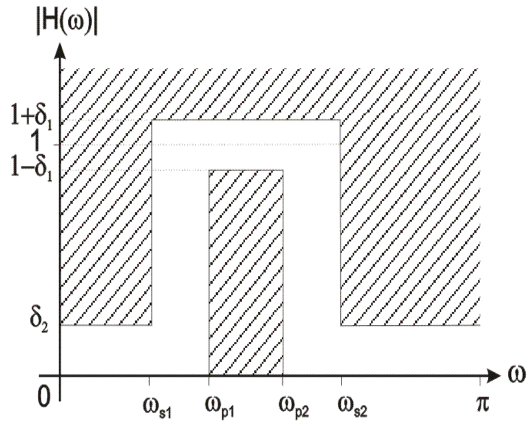


Fig. 8a. Band-pass digital filter specification.

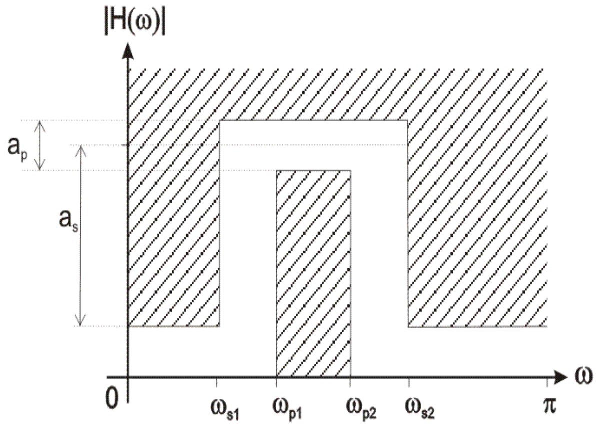


Fig. 8b. Band-pass digital filter specification.

Figure 9 illustrates a band-stop digital filter specification.

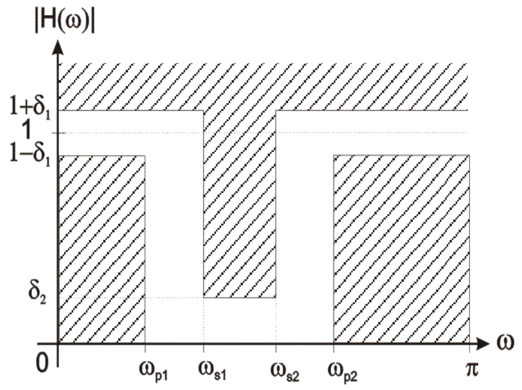


Fig. 9a. Band-stop digital filter specification.

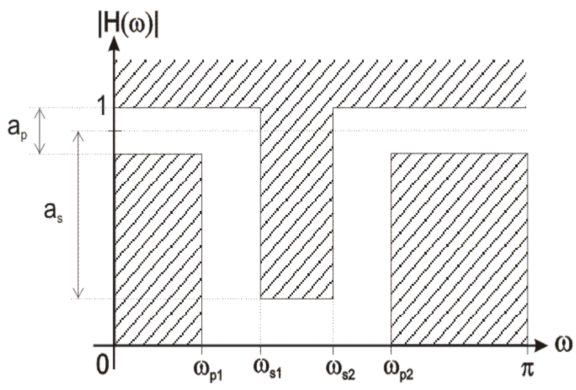


Fig. 9b. Band-stop digital filter specification.

Prototype Filter Types

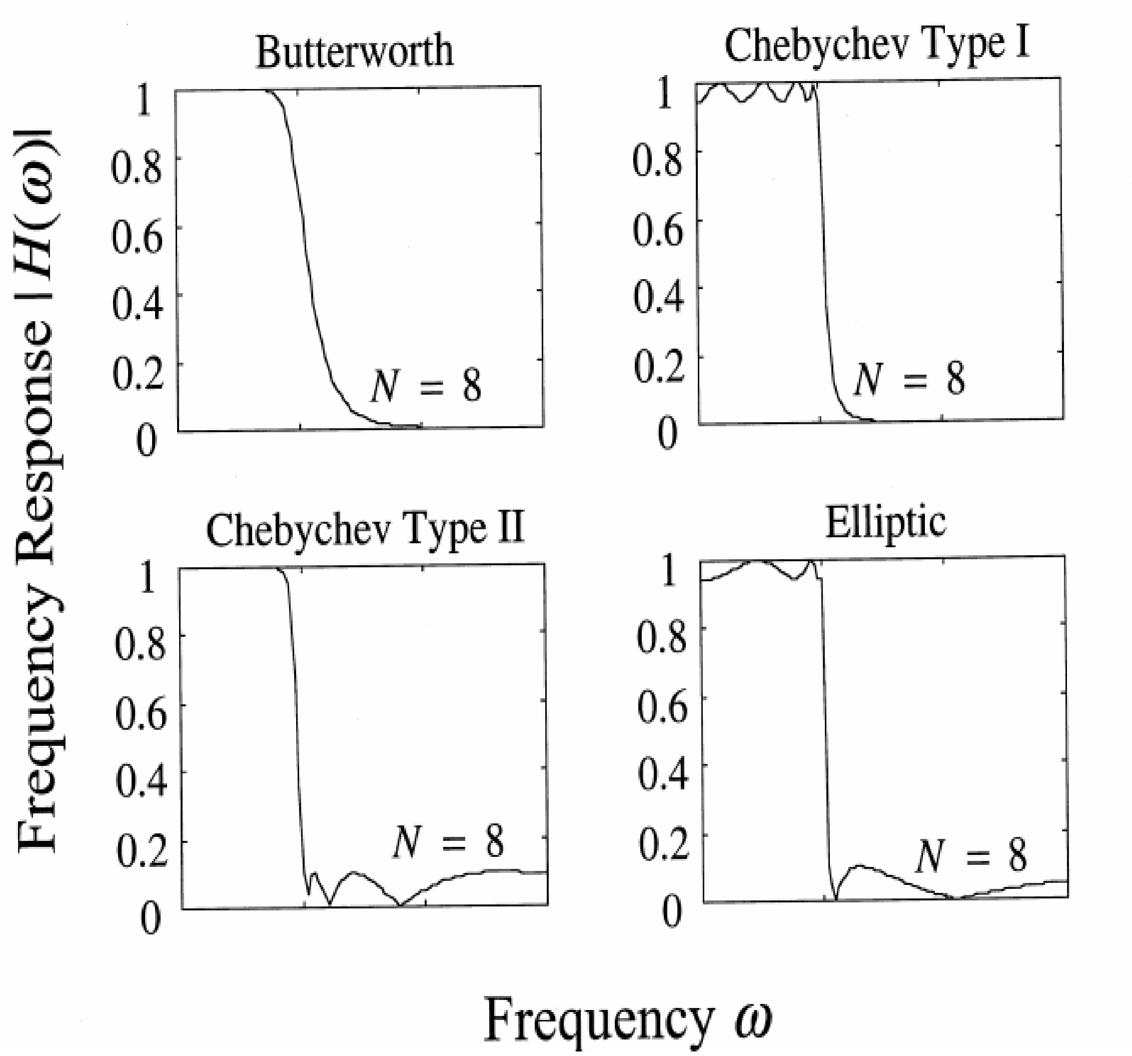


Fig. 10: Prototype Filter Types.

Butterworth Filters

Butterworth filters are maximally flat IIR filters. The flatness in the passband and stopband causes the transition band to be very wide. Large orders are required to obtain filters with narrow transition widths.

Chebyshev Type I Filters

Chebyshev type I filters attain smaller transition widths than Butterworth filters of the same order by allowing for passband ripple. The stopband of a Chebyshev type I filter is, as with Butterworth filters, maximally flat. For a given filter order, the tradeoff is thus between passband ripple and transition width.

Chebyshev Type II Filters

Chebyshev type II filters have maximally flat passband and equiripple stopband. Since extremely large attenuations are typically not required, we may be able to attain the required transition width with a relatively small order by allowing for some stopband ripple.

Elliptic Filters

Elliptic filters generalize Chebyshev and Butterworth filters by allowing for ripple in both the passband and the stopband. As ripples are made smaller, elliptic filters can approximate arbitrarily close the magnitude and phase response of either Chebyshev or Butterworth filters. The extra degrees of freedom allow elliptic filters to have the smallest order for a given transition width.

DIGITAL FILTER DESIGN PROCESS

Design filter design involves the following basic steps:

1) Determine a desired response or a set of desired responses (eg., a desired magnitude response and/or a desired phase response).

2) Select a class of filters for approximating the desired response(s) (eg., linear-phase FIR filter or IIR filters being implementable as a parallel connection of two allpass filters).

3) Establish a criterion of “goodness” for the response(s) of a filter in the selected class compared to the desired response(s).

4) Develop a method for finding the best member in the filter class.

5) Synthesize the best filter using a proper structure and a proper implementation form, for example using a computer program, a signal processor, or a VLSI chip.

6) Analyze the filter performance.

In most cases, the desired response is the given magnitude response or the given phase (delay) response or both. The desired magnitude response is usually specified by determining the frequency region(s) where the input signal components should be preserved and the region(s) where the signal components should be rejected. The phase response, in turn, is often desired to be linear in those frequency intervals where the signal components are preserved. In certain cases, time-domain conditions may be included, for example, in the design of Nyquist filter where some of the impulse-response values are restricted to be zero value. There are also applications where constraints on the step response are imposed.

The second step consists of determining a proper class of filters to approximate the given response(s). First, it must be decided whether to use FIR filters r IIR filters. After this, a proper class of FIR and IIR filter is selected. For many computationally efficient or low - sensitivity FIR or IIR filter structures, there are constraints on the transfer function. In these cases, the design of the transfer function and the filter implementation cannot be separated, and the desired filter structure determines the class of filters under consideration.

In order to find the best member in the selected filter class, an error measure is needed by which the nearness of the approximating response(s) to the given response(s) is determined.

The fourth step is to find or develop a method for finding this best member. The fifth step involves synthesizing the filter designed at the previous step. The final step is to test whether the resulting filter meets all the given criteria. The diagram below shows the block diagram of FIR and IIR filters:

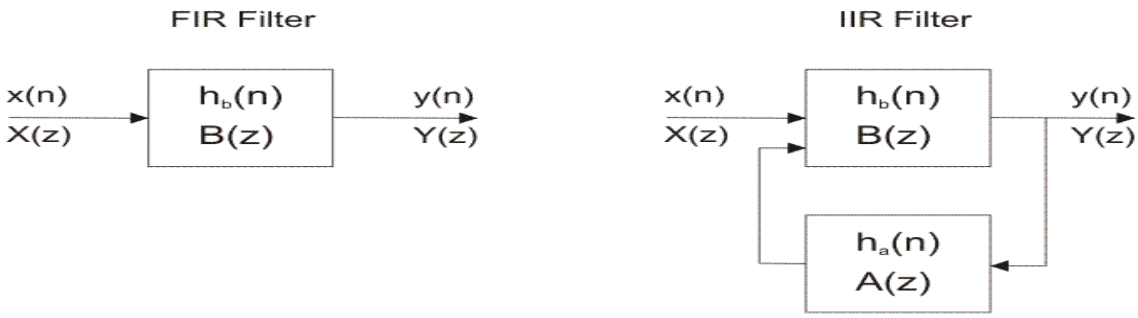


Fig. 11 Block Diagrams of FIR and IIR Filters.

FIR VS. IIR FILTERS

❖ FIR Filter Equation

$$y[n] = a_0 x[n] + a_1 x[n-1] + \dots + a_{M-1} x[n-(M-1)]$$

❖ IIR Filter Equation

$$\begin{aligned} &y[n] = a_0 x[n] + a_1 x[n-1] + \dots + a_{M-1} x[n-(M-1)] \\ &- b_1 y[n-1] - b_2 y[n-2] - \dots - b_N y[n-N] \end{aligned}$$

IIR filters can generally achieve given desired response with less computation than FIR filters It is easier to approximate arbitrary frequency response characteristics with FIR filters, including exactly linear phase.

SELECTION OF FILTER TYPE

The transfer function H(z) meeting the frequency response specifications should be a causal transfer function. For IIR digital filter design, the IIR transfer function is a real rational function of: $z^{-1} = H(z) = \dfrac{p_0 + p_1 z^{-1} + p_2 z^{-2} + \dots + p_M z^{-M}}{d_0 + d_1 z^{-1} + d_2 z^{-2} + \dots + d_N z^{-N}}$. H(z) must be a stable transfer function and must be of lowest order N for reduced computational complexity. For FIR digital filter design, the FIR transfer function is a polynomial in z^{-1} with real coefficients: $H(z) = \sum_{n=0}^{N} h[n] z^{-n}$. For reduced computational complexity, degree N of $H(z)$ must be as small as possible. If a linear phase is desired, the filter coefficients must satisfy the constraint: $h[n] = \pm h[N-n]$.

FIR DIGITAL FILTER DESIGN: BASIC APPROACHES

FIR filters are digital filters with finite impulse response. They are also known as non-recursive digital filters as they do not have the feedback (a recursive part of a filter), even though recursive algorithms can be used for FIR filter realization. FIR filters can be designed using different methods, but most of them are based on ideal filter approximation. The objective is not to achieve ideal characteristics, as it is impossible anyway, but to achieve sufficiently good characteristics of a filter. The transfer function of FIR filter approaches the ideal as the filter order increases, thus increasing the complexity and amount of time needed for processing input samples of a signal being filtered. FIR filter design is based on a direct approximation of the specified magnitude response, with the often added requirement that the phase be linear. The design of an FIR filter of order N may be accomplished by finding either the length $-(N+1)$ impulse response samples $\{h[n]\}$ or the $(N+1)$samples of its frequency response $H(e^{j\omega})$. Three commonly used approaches to FIR filter design are:

(1) Windowed Fourier series approach,

(2) Frequency sampling approach,

(3) Computer-based optimization methods.

ADVANTAGES IN USING FIR FILTER

In many digital signal processing applications, FIR filters are preferred over their IIR counterparts. The main advantages of the FIR filter designs over their IIR equivalents are the following:

1) FIR filters with exactly linear phase can easily be designed.

2) Filter structure is always stable with quantized coefficients.

FIR filters realized nonrecursively are inherently stable and free of limit cycle oscillations when implemented on a finite wordlengh digital system.

3) Excellent design methods are available for various kinds of FIR filters with arbitrary specifications.

4) They can be realized efficiently in hardware.

5) The filter startup transients have finite duration.

6) The output noise due to multiplication roundoff errors in an FIR filter is usually very low and the sensitivity to variations in the filter coefficients is also low.

DISADVANTAGES IN USING FIR FILTER

The main disadvantage of conventional FIR filter designs is that they require, especially in applications demanding narrow transition bands, considerable more arithmetic operations and hardware components such as multipliers, adders, and delay elements than do comparably IIR filters. As the transitions bandwidth of an FIR filter is made narrower, the filter order, and correspondingly the arithmetic complexity, increases inversely proportional to this its width. This makes the implementation of narrow transmission band for FIR filters very costly. The cost of implantation of FIR filters can, however, be reduced by using multiplier-efficient realizations, fast convolution algorithm and multirates filtering. Other disadvantages include:

1) The delay of these filters is often much greater than for an equal performance IIR filters.

2) Order of an FIR filter, in most cases, is considerably higher than the order of an equivalent IIR filter meeting the same specifications.

3) FIR filter has thus higher computational complexity.

LOW PASS FIR FILTER DESIGN

The ideal lowpass filter is one that leaves unchanged all frequency components of a signal below a designated cutoff frequency, ω_c, and rejects all components above ω_c. Because the impulse response required to implement the ideal lowpass filter is infinitely long, it is impossible to design an ideal FIR lowpass filter. Finite length approximations to the ideal impulse response lead to the presence of ripples in both the passband $(\omega < \omega_c)$ and the stopband $(\omega > \omega_c)$ of the filter, as well as to a nonzero transition width between passband and stopband. Both the passband/stopband ripples and the transition width are undesirable but unavoidable deviations from the response of an ideal lowpass filter when approximated with a finite impulse response. These deviations are depicted in the figure 12 below:

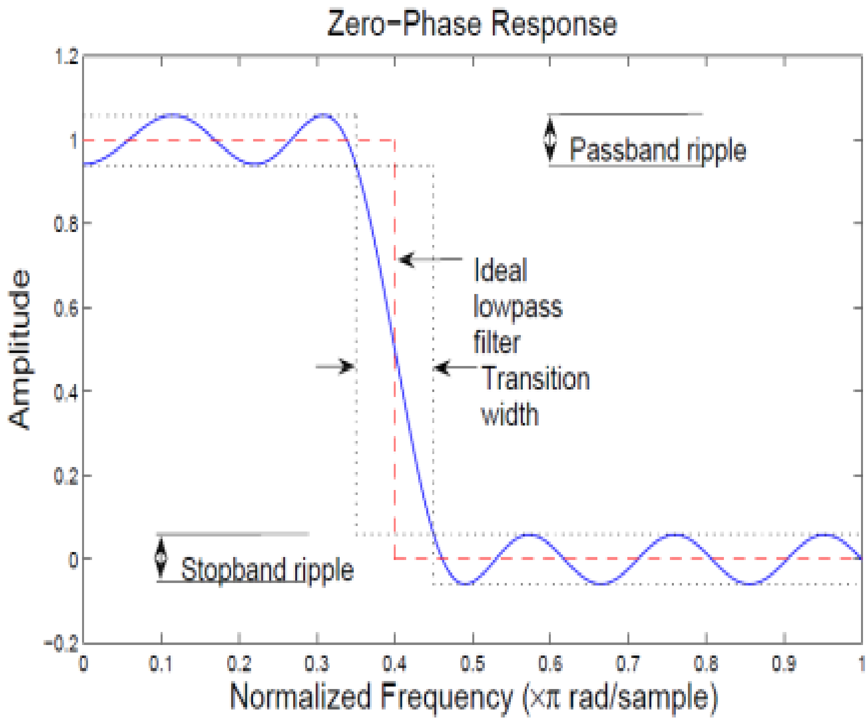


Fig.12: Deviations in FIR.

Practical FIR designs typically consists of filters that meet certain design specifications, i.e., that have a transition width and maximum passband and stopband ripples that do not exceed allowable values. In addition, one must select the filter order, or, equivalently, the length of the truncated impulse response.

FIR FILTER DESIGN USING WINDOW FUNCTIONS

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

- Defining filter specifications;
- Specifying a window function according to the filter specifications;
- Computing the filter order required for a given set of specifications;
- Computing the window function coefficients;
- Computing the ideal filter coefficients according to the filter order;
- Computing FIR filter coefficients according to the obtained window function and ideal filter coefficients;
- 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 $(\omega c, \omega c1, \omega c2)$ transition width and stopband 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:

$$h[n] = w[n].h_d[n].$$

Lastly, the transfer function of designed filter will be found by transforming impulse response via Fourier transform: $H(e^{j\omega}) = \sum_{n=0}^{N} h[n].e^{-jn\omega}$

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, re-estimate the window function coefficients and ideal filter frequency samples, multiply them in order to obtain the frequency response of designed filter and re-estimate 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 re-estimate the filter frequency coefficients after that. For the sake of precise estimates, the filter order should be decreased or increased by 1. FIR filter transfer function can be expressed as: $H(Z) = \frac{Y(Z)}{X(Z)} = \sum_{n=0}^{N-1} h[n].\, z^{-n}$.

The frequency response realized in the time domain is of more interest for FIR filter realization (both hardware and software). The transfer function can be found via the z-transform of a FIR filter frequency response. FIR filter output samples can be computed using the following expression:

$$y[n]=\sum_{k=0}^{N-1} h[k].\,\mathrm{x}[\mathrm{n}-\mathrm{k}]$$

where:

- x[k] are FIR filter input samples;
- h[k] are the coefficients of FIR filter frequency response; and
- y[n] are FIR filter output samples.

A good property of FIR filters is that they are less sensitive to the accuracy of constants than IIR filters of the same order.

LOWPASS FILTER SPECIFICATIONS

1. Choose a pass band edge frequency for design that is midway through the transition width.

2. Find the impulse response $\mathrm{h_L}[\mathrm{n}]$ for the ideal low pass filter with this pass band edge frequency.

3. Choose a window w[n] that gives adequate stop band attenuation, and determine the number of terms N needed to give the required transition width. An odd number should be chosen, so that the impulse response will be perfectly symmetrical around its middle point.

4. Find the impulse response $\mathrm{h}[\mathrm{n}]=\mathrm{h_L}[\mathrm{n}]\omega[\mathrm{n}]$ for the FIR filter.

5. Shift the impulse response to the right by $(\mathrm{N}-1)/2$ samples.

INFINITE IMPULSE RESPONSE (IIR) FILTER DESIGN-(OVERVIEW)

- IIR digital filter designs are base on established methods for designing analog filters.
- Approach is generally limited to frequency selective filters with ideal passband/stopband characteristics
- Basic filter type is low pass
- Achieve highpass or bandpass via transformations.
- Achieve multiple stop/pass bands by combining multiple filters with single pass band.

IIR FILTER DESIGN STEPS

- Choose prototype analog filter family
  - Butterworth,
  - Chebychev Type I or II,
  - Elliptic.
- Choose analog-digital transformation method
  - Impulse Invariance.
  - Bilinear Transformation.
  - Transform digital filter specifications to equivalent analog filter specifications.
  - Ability to design lowpass IIR filters according to predefined specifications based on analog filter theory and analog-to-digital filter transformation.
  - Transform analog filter to digital filter.
  - Perform frequency transformation to achieve highpass or bandpass filter, if desired.
  - Ability to construct frequency-selective IIR filters based on a lowpass IIR filter.

MOST COMMON APPROACH TO IIR FILTER DESIGN

1. Convert the digital filter specifications into an analog prototype lowpass filter specifications

2. Determine the analog lowpass filter transfer function $H_a(s)$

3. Transform $H_a(s)$ into the desired digital transfer function $G(z)$

This approach has been widely used for the following reasons:

(a) Analog approximation techniques are highly advanced

(b) They usually yield closed-form solutions

(c) Extensive tables are available for analog filter design

(d) Many applications require digital simulation of analog systems

An Analog transfer function is denoted as $H_a(s) = \frac{P_a(s)}{D_a(s)}$, where the subscript "a" specifically indicates the analog domain. A digital transfer function derived from $H_a(s)$ shall be denoted as $G(z) = \frac{P(z)}{D(z)}$. The basic idea behind the conversion of $H_a(s)$ into $G(z)$ is to apply a mapping from the *s*-domain to the *z*-domain so that essential properties of the analog frequency response are preserved. Thus mapping function should be such that:

1. Imaginary ($j\Omega$) axis in the *s*-plane be mapped onto the unit circle of the *z*-plane.
2. A stable analog transfer function be mapped into a stable digital transfer function. We therefore seek to design IIR filter based on the above advantages.

IIR FILTER DESIGN

One of the drawbacks of FIR filters is that they require a large filter order to meet some design specifications. If the ripples are kept constant, the filter order grows inversely proportional to the transition width. By using feedback, it is possible to meet a set of design specifications with a far smaller filter order. This is the idea behind IIR filter design. The term "infinite impulse response" (IIR) stems from the fact that, when an impulse is applied to the filter, the output never decays to zero. IIR filters tend to be used when computational resources are at a premium. However, stable, causal IIR filters cannot have perfectly linear phase. IIR filters thus tend to be avoided when linearity of phase is a requirement. Another important reason for using IIR filters is their small group delay relative to FIR filters, which results in a shorter transient response.

Bilinear Transformation:

$$s = \frac{2}{T}\left(\frac{1-Z^{-1}}{1+Z^{-1}}\right),\ Z = \frac{1+S}{1-S}$$

The above transformation maps a single point in the s-plane to a unique point in the z-plane and vice-versa. Relation between $G(z)$ and $H_a(s)$ is then given by:

$G(z) = H_a(s)\Big|_{s=\frac{2}{T}\left(\frac{1-Z^{-1}}{1+Z^{-1}}\right)}$. Bilinear transformation digital filter design consists 3 steps:

1. Develop the specifications of $H_a(s)$ by applying the inverse bilinear transformation to specifications of $G(z)$,

2. Design $H_a(s)$,

3. Determine $G(z)$ by applying bilinear transformation to $H_a(s)$.

As a result, the parameter T has no effect on $G(z)$ and T=2 is chosen for convenience. The inverse bilinear transformation for T=2 is $Z = \frac{1+S}{1-S}$

For $s = \sigma_0 + j\Omega_0$

$$z = \frac{(1+\sigma_0)+j\Omega_0}{(1-\sigma_0)-j\Omega_0} \Rightarrow |z|^2 = \frac{(1+\sigma_0)^2 + j\Omega_0{}^2}{(1-\sigma_0)^2 + j\Omega_0{}^2}$$

$\sigma_0 = 0 \rightarrow |z| = 1$

And so, $\sigma_0 < 0 \rightarrow |z| < 1$

$\sigma_0 > 0 \rightarrow |z| > 1$

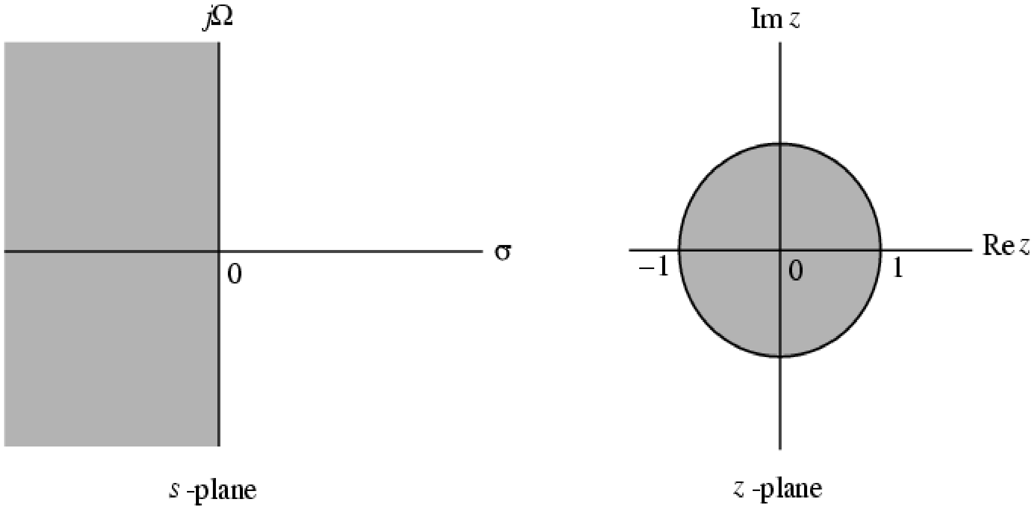


Fig. 13: Mapping of the s-plane into the z-plane.

For $z = e^{j\omega}$ with T = 2, we have $j\Omega = \dfrac{1 - e^{-j\omega}}{1 + e^{-j\omega}} = j\tan\left(\omega/2\right)$ Or $\Omega = \tan\left(\omega/2\right)$

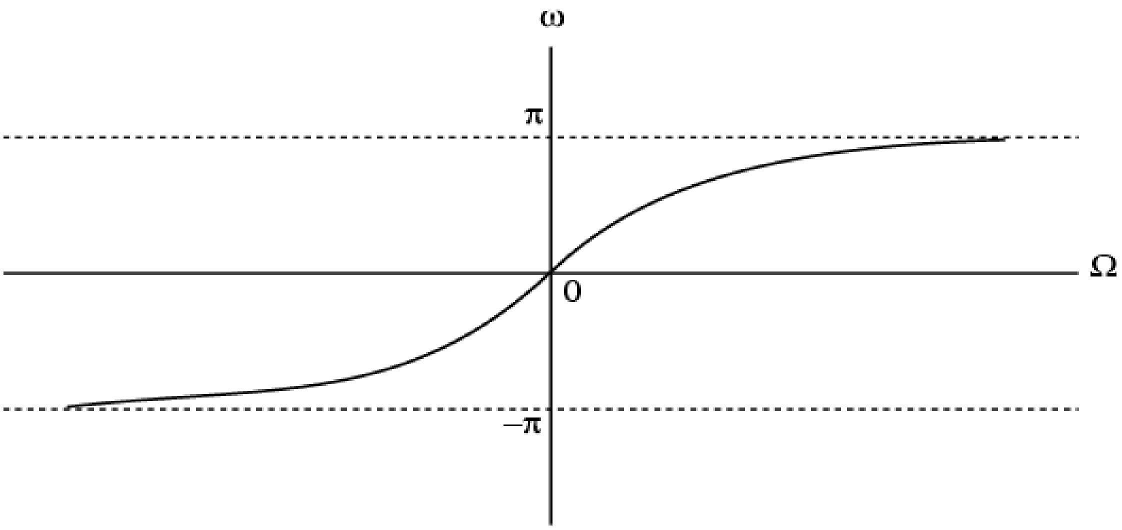


Fig. 14: Mapping of the angular analog frequency to the angular digital frequencies via the bilinear transformation.

NON LINEAR MAPPING

Mapping is highly nonlinear. Complete negative imaginary axis in the *s*-plane from $\Omega = -\infty$ to $\Omega = 0$ is mapped into the lower half of the unit circle in the *z*-plane from $z = -1$ to $z = 1$. Complete positive imaginary axis in the *s*-plane from $\Omega = 0$ to $\Omega = \infty$ is mapped into the upper half of the unit circle in the *z*-plane from z = 1 to z **= - 1.** Nonlinear mapping introduces a distortion in the frequency axis called frequency warping.

Effect of warping is shown below:

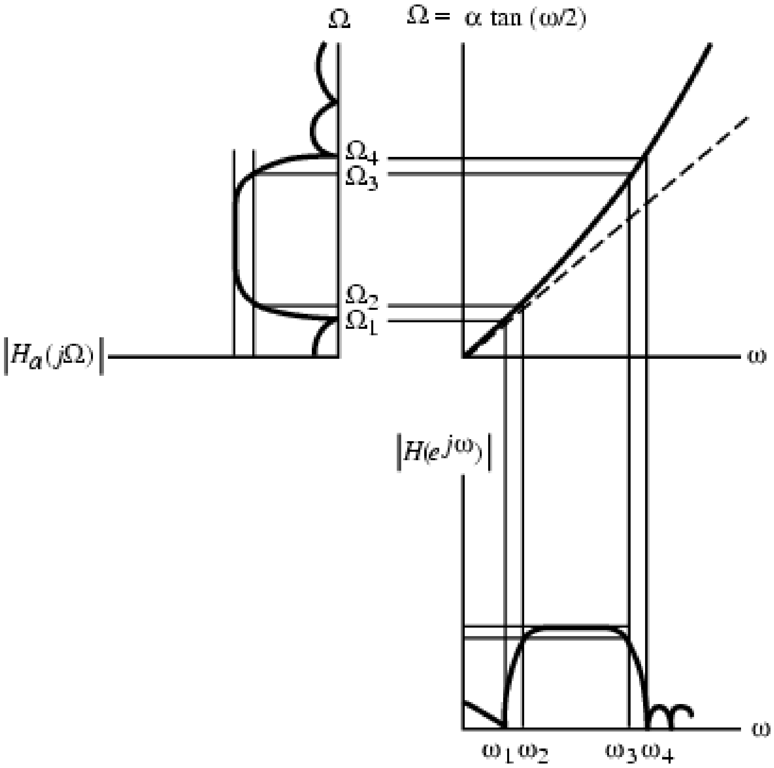


Fig. 15: Effect of Frequency warping.

STEPS IN THE DESIGN OF IIR HIGHPASS, BANDPASS, AND BANDSTOP DIGITAL FILTERS

Transformation can be used only to design digital filters with prescribed magnitude response with piecewise constant values. Transformation does not preserve phase response of analog filter.

First Approach:

1. Prewarp digital frequency specifications of desired digital filter $G_D(z)$ to arrive at frequency specifications of analog filter $H_D(s)$ of same type

2. Convert frequency specifications of $H_D(s)$ into that of prototype analog lowpass filter $H_{LP}(s)$

3. Design analog lowpass filter $H_{LP}(s)$

4. Convert $H_{LP}(s)$ into $H_D(s)$ using inverse frequency transformation used in Step 2

5. Design desired digital filter $G_D(z)$ by applying bilinear transformation to $H_D(s)$

Second Approach:

1. Prewarp digital frequency specifications of desired digital filter $G_D(z)$ to arrive at frequency specifications of analog filter $H_D(s)$ of same type

2. Convert frequency specifications of $H_D(s)$ into that of prototype analog lowpass filter $H_{LP}(s)$

3. Design analog lowpass filter $H_{LP}(s)$

4. Convert $H_{LP}(s)$ into an IIR digital transfer function $G_{LP}(z)$ using bilinear transformation

5. Transform $G_{LP}(z)$ into the desired digital transfer function $G_D(s)$

SPECTRAL TRANSFORMATIONS

To transform $G_L(z)$ a given lowpass transfer function to another transfer $G_D\left(\hat{z}\right)$ function that may be a lowpass, highpass, bandpass or bandstop filter, Z^{-1} has been used to denote the unit delay in the prototype lowpass filter $G_L(z)$ and Z^{-1} to denote the unit delay in the transformed filter $G_D\left(\hat{z}\right)$ to avoid confusion.

$$\text{Unit circles in z and z - planes are defined by } z = e^{j\omega} \text{ and } \hat{z} = e^{j\hat{\omega}}$$

$$\text{Transformation from z to } \hat{z} \text{– domain is given by } z = F\left(\hat{z}\right)$$

$$\text{Then } G_D\left(\hat{z}\right) = G_L\left\{F\left(\hat{z}\right)\right\}, \text{ From } z = F\left(\hat{z}\right), \text{ thus } |z| = \left|F\left(\hat{z}\right)\right|$$

$$\text{Hence} \left|F\left(\hat{z}\right)\right| \begin{cases} >1, & \text{if } |z|>1 \\ =1, & \text{if } |z|=1 \\ <1, & \text{if } |z|<1 \end{cases}$$

$$\text{Therefore } \frac{1}{F\left(\hat{z}\right)} \text{ must be a stable all pass function : } \frac{1}{F\left(\hat{z}\right)} = \pm\prod_{\ell=1}^{L}\left(\frac{1-\alpha_{*\ell}\,\hat{z}}{\hat{z}-\alpha_{\ell}}\right),\ |\alpha_{\ell}|<1$$

LOWPASS TO LOWPASS SPECTRAL TRANSFORMATION

To transform a lowpass filter $G_L(z)$ with cut-off frequency ω_c to another lowpass filter $G_D\left(\hat{z}\right)$ with cut-off frequency $\hat{\omega}_c$, the transformation is:

$$z^{-1} = \frac{1}{F\left(\hat{z}\right)} = \frac{1-\alpha\hat{z}}{\hat{z}=\alpha}, \text{ on the unit circle, we have } e^{-j\omega} = \frac{e^{-j\hat{\omega}}=\alpha}{1-\alpha e^{-j\omega}}$$

$$\text{which yields } \tan\left(\frac{\omega}{2}\right) = \left(\frac{1+\alpha}{1+\alpha}\right)\tan\frac{\hat{\omega}}{2}. \text{ Solving we get } \alpha = \frac{\operatorname{Sin}\left(\left(\omega_c-\hat{\omega}_c\right)/2\right)}{\operatorname{Sin}\left(\left(\omega_c+\hat{\omega}_c\right)/2\right)}$$

ADVANTAGES OF USING IIR

1) They typically meet a given set of specifications with a much lower filter order than a corresponding FIR filter.

2) The IIR filter is better than the FIR in that it can produce the same response using fewer delay blocks (so uses less of the processor resources).

3) IIR filters are well suited for applications that require no phase information, for example, for monitoring the signal amplitudes. FIR filters are better suited for applications that require a linear phase response.

DISADVANTAGES OF USING IIR

1) IIR filters can become unstable.

2) IIR filters have a feedback loop so they will accumulate rounding and noise error.

3) IIR filters usually have a non-linear phase response.

CONCLUSION

From the explanation and design specifications above, it is seen that both FIR and IIR can be realized if their design specifications are provided. It could to be pointed out that the constant group delay of analog filters does not transform to a constant group delay of the IIR filter obtained by the bilinear transformation. Using the poles and zeros of a classical lowpass prototype filter in the continuous (Laplace) domain, we obtained a digital filter through frequency transformation and filter discretization. The specifications and design method described in this paper is basically geared towards the design of IIR digital filter, since the primary advantage of IIR filters over FIR filters is that they typically meet a given set of specifications with a much lower filter order than a corresponding FIR filter. Although IIR filters have nonlinear phase, data processing within MATLAB software is commonly performed, that is, the entire data sequence is available prior to filtering. This allows for a noncausal, zero-phase filtering approach, which eliminates the nonlinear phase distortion of an IIR filter.

REFERENCES

[1] Y. Neuvo, C. Dong, and S.K. Mitra, "Interpolated finite impulse response filters," in IEEE Transactions on Acoustics, Speech, and Signal Processing,vol. 32, no. 3, pp. 563-570, June 1984.

[2] http://www.mathworks.com/help/toolbox/signal/f4-1046.htm.

[3] Sanjit .K. Mitra, Digital Signal Processing: A Computer Based Approach, 7nd Edition McGraw Hill 2000.12, pages 423-498.

[4] A.G. Constantinides, An introductory lecture to Digital Signal Processing & Digital Filters, Imperial College London, 2007.

[5] Dzhankhotov V., Hybrid LC filter for power electronic drives: Theory and Implementation, 2009.

[6] Matthaei, Young, Jones Microwave Filters, Impedance-Matching Networks, and Coupling Structures McGraw-Hill 1964

[7] Zverev, Anatol, I (1969). Handbook of Filter Synthesis. John Wiley & Sons. ISBN 0-471-98680-1. Catalog of passive filter types and component values. The Bible for practical electronic filter design.

[8] Williams, Arthur B & Taylor, Fred J (1995). Electronic Filter Design Handbook. McGraw-Hill. ISBN 0-07-070441-4.

[9] A. Antoniou, New improved method for the design of weighted-Chebyshev, non-recursive, digital filters. IEEE Trans. Circuits Syst. 129,1-10(1982); and Erratum: ibid. pg.107(1982).

[10] R. Boite and H. Leich, Comments on ‘A fast procedure to design equiripple minimum-phase FIR filters.’ IEEE Trans. Syst. CAS-31,503-504(1984).