LECTURE HANDOUTS **L-1**

ECE **II/IV**

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu.Prof./ECE

Unit : FOURIER ANALYSIS OF DISCRETE TIME SIGNALS Date of Lecture :

Topic of Lecture: Introduction to DSP

Introduction :

- DSP manipulates different types of signals with the intention of filtering, measuring, or compressing and producing analog signals.
- An analog-to-digital converter is needed in the real world to take analog signals (sound, light, pressure, or temperature) and convert them into 0's and 1's for a digital format

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)

1.Signals and System
2.Fourier Transform
3.Z-Transform

Introduction

Signal:

A *signal* is defined as any physical quantity that varies with time, space, or any other in dependent variable or variables. Mathematically, we describe a signal as a function of one or mo re independent variables. For example, the functions s(t)= 5t
describe a signal, one that varies linearly with the in d e p e n d e n t variable *t* (time).

$$s(x, y) = 3x + 2xy + 10y^2$$

This function describes a signal of two in dependent variables *x* and y that could represent the two spatial coordinates in a p lane.

System:

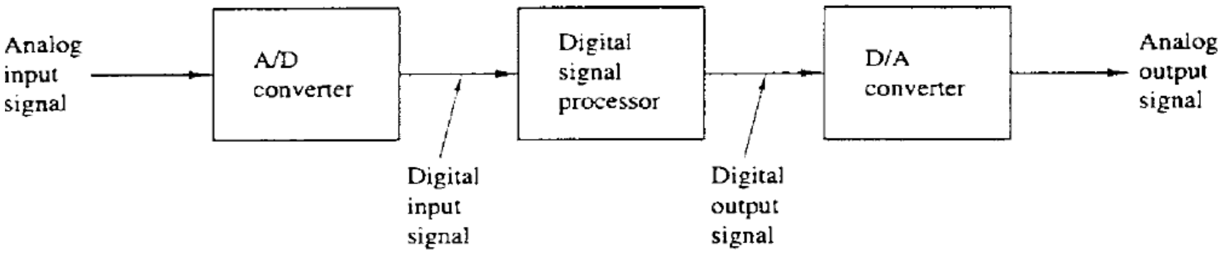
A *system* may also be defined as a physical device that performs an operation on a signal. For ex ample, a filter used to reduce the noise and interference corrupting desired in formation bearing signal is called a system .

signal processing:

When we pass a signal through a system , as in filtering, we say that we have processed the signal. In this case the processing of the signal involves filtering the noise and interference from the desired signal. If the operation on the signal is n o n linear, the system is said to be nonlinear, and so forth . Such operations are usually referred to as *signal processing.*

Analog signal processing:

Digital signal processing:



Advantages of Digital over Analog Signal Processing :

- A digital programmable system allows s flexibility in re configuring the digital signal processing operations simply by changing the program.
- A digital system provides much better control of accuracy.
- Digital signals are easily stored on magnetic media (tape or disk) without deterioration or loss of signal fidelity beyond that introduced in the A /D conversion.
- Digital implementation of the signal processing system is cheaper than analog signal processing.

Limitations:

- One practical limitation is the speed of operation of A /D converters and digital signal processors. We shall see that signals having extremely wide band widths require fast-sampling
- -rate A /D converters and fast digital signal processors. Hence there are analog signals with large bandwidths for which a digital processing approach is beyond the state of the art of digital hardware.

Video Content / Details of website for further learning (if any):
https://www.allaboutcircuits.com/technical-articles/an-introduction-to-digital-signal-processing/

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg:210-212

Course Faculty

Verified by HOD

LECTURE HANDOUTS

L-2

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu.Prof./ECE

Unit : FOURIER ANALYSIS OF DISCRETE TIME SIGNALS Date of Lecture :

Topic of Lecture: Introduction to DFT

Introduction : (Maximum 5 sentences)

- The discrete Fourier transform (DFT) is a fundamental transform in digital signal processing, with applications in frequency analysis, fast convolution, image processing, etc.
- DFT is to compute the frequency responses of filters, to implement convolution, and spectral estimation

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

1.Signals and System
2.Fourier Transform
3.Z-Transform

Detailed content of the Lecture:

The discrete Fourier transform (DFT) derived from the Fourier series The exponential Fourier series of a continuous time periodic signal x(t) with fundamental period T0 is given by the synthesis equation

$$x(t) = \sum_{k=-\infty}^{\infty} X_k \, e^{j2\pi k F_0 t} \qquad \rightarrow (1)$$

where the Fourier coefficients X_k are given by the analysis equation

$$X_k = \frac{1}{T_0}\int_{T_0} x(t)\, e^{-j2\pi k F_0 t}\, dt \qquad \rightarrow (2)$$

with the fundamental frequency F_0 and the period T_0 related by F_0 (Hz) = $1/T_0$ (sec).
To obtain finite-sum approximations for the above two equations, consider the analog periodic signal x(t) shown in Figure and its sampled version $x_s(nT)$. Using $x_s(nT)$, we can approximate the integral for Xk by the sum

$$X_k = \frac{1}{T_0}\sum_{n=0}^{N-1} x_s(nT)\, e^{-j2\pi k F_0 nT}\, T, \qquad k = 0, 1, \ldots, N-1$$

$$= \frac{1}{N}\sum_{n=0}^{N-1} x(n)\, e^{-j2\pi k n/N}, \qquad k = 0, 1, \ldots, N-1$$

where we used the relation $F_0T = 1/N$, and approximated dt (or Δt) by T, and have used the shorthand notation x(n) = xs(nT). (This procedure is similar to that used in a typical introduction to integral calculus). Discrete time domain signal is converting into discrete frequency domain

$$X_k = \sum_{n=0}^{N-1} x_n e^{-j2\pi kn/N}, \quad k = 0, 1, \ldots, N-1 \qquad \rightarrow (3)$$

And the inverse DFT is defined as

$$x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k e^{\,j2\pi kn/N}, \qquad n = 0, 1, \ldots, N-1 \qquad \rightarrow (4)$$

Find the DFT of the **unit sample** $x(n) = \{1, 0, 0, 0\}$.

Solution The number of samples is $N = 4$. The DFT is given by

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{-jk2\pi n/N}, \qquad 0 \le k \le N-1$$

$$= \sum_{n=0}^{4-1} x(n)\, e^{-jk2\pi n/4}, \qquad 0 \le k \le 3$$

$$= \sum_{n=0}^{3} x(n)\, e^{-jk2\pi n/4}, \qquad k = 0, 1, 2, 3$$

$$\sum_{n=0}^{3} x(n)\, e^{-jk2\pi n/4}, \; k = 0, 1, 2, 3$$

$k = 0$	$X(0) = \sum_{n=0}^{3} x(n)\, e^{-j0.2\pi n/4} = \sum_{n=0}^{3} x(n).1 = \sum_{n=0}^{3} x(n)$ $= x(0) + x(1) + x(2) + x(3) = 1 + 0 + 0 + 0 = 1$
$k = 1$	$X(1) = \sum_{n=0}^{3} x(n)\, e^{-j1.2\pi n/4} = \sum_{n=0}^{3} x(n)\, e^{-j\pi n/2} = x(0)\, e^{-j0} = 1 \,.\, 1 = 1$
$k = 2$	$X(2) = \sum_{n=0}^{3} x(n)\, e^{-j2.2\pi n/4} = \sum_{n=0}^{3} x(n)\, e^{-j\pi n} = x(0)\, e^{-j0} = 1 \,.\, 1 = 1$
$k = 3$	$X(3) = \sum_{n=0}^{3} x(n)\, e^{-j3.2\pi n/4} = \sum_{n=0}^{3} x(n)\, e^{-j3\pi n/2} = x(0)\, e^{-j0} = 1 \,.\, 1 = 1$

The DFT of X(k)={1, 1, 1, 1}

Find the inverse discrete Fourier transform of $X(k) = \{3, (2+j), 1, (2-j)\}$.

Solution The number of samples is $N = 4$. The IDFT is given by

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)\, e^{j2\pi kn/N}, \qquad n = 0, 1, \ldots, N-1$$

$$= \frac{1}{4}\sum_{k=0}^{3} X(k)\, e^{j2\pi kn/4},$$

$$= \frac{1}{4}\sum_{k=0}^{3} X(k)\, e^{jkn\pi/2}, \qquad 0 \le n \le 3$$

The calculations for $\{x(n), n = 0 \text{ to } 3\}$ are shown in table below.

$$x(n) = \frac{1}{4}\sum_{k=0}^{3} X(k)\, e^{jkn\pi/2}$$

$n = 0$	$x(0) = \frac{1}{4}\sum_{k=0}^{3} X(k) e^{jk0\pi/2} = \frac{1}{4}\sum_{k=0}^{3} X(k)$ $= (1/4)\{X(0) + X(1) + X(2) + X(3)\} = (1/4)\{3 + 2+j + 1 + 2-j\} = 2$
$n = 1$	$x(1) = (1/4)\sum_{k=0}^{3} X(k) e^{jk1\pi/2} = (1/4)\sum_{k=0}^{3} X(k)\left(e^{j\pi/2}\right)^k = (1/4)\sum_{k=0}^{3} X(k)(j)^k$ $= (1/4)\{X(0)(j)^0 + X(1)(j)^1 + X(2)(j)^2 + X(3)(j)^3\}$ $= (1/4)\{3 . 1 + (2+j) . j + 1 . (-1) + (2-j) . (-j)\} = 0$
$n = 2$	$x(2) = (1/4)\sum_{k=0}^{3} X(k) e^{jk2\pi/2} = (1/4)\sum_{k=0}^{3} X(k) e^{jk\pi} = (1/4)\sum_{k=0}^{3} X(k)(-1)^k$ $= (1/4)\{X(0)(1) + X(1)(-1) + X(2)(1) + X(3)(-1)\}$ $= (1/4)\{X(0) - X(1) + X(2) - X(3)\}$ $= (1/4)\{3 - (2+j) + 1 . (-1) - (2-j)\} = 0$
$n = 3$	$x(3) = (1/4)\sum_{k=0}^{3} X(k) e^{jk3\pi/2} = (1/4)\sum_{k=0}^{3} X(k)\left(e^{j3\pi/2}\right)^k = (1/4)\sum_{k=0}^{3} X(k)(-j)^k$ $= (1/4)\{3 . 1 + (2+j). (-j) + 1 . (-1) + (2-j). j\}$ $= (1/4)\{3 - j2 + 1 - 1 + j2 + 1\} = 1$

The IDFT of X(n)={2, 0, 0, 1}

Video Content / Details of website for further learning (if any):
https://www.tutorialspoint.com/digital_signal_processing/dsp_discrete_fourier_transform_introduction.htm

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg:399 -401

Course Faculty

Verified by HOD

LECTURE HANDOUTS

L-3

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu.Prof./ECE

Unit : FOURIER ANALYSIS OF DISCRETE TIME SIGNALS Date of Lecture :

Topic of Lecture: Properties of Discrete Fourier Transform

Introduction :

- The properties of the DFT (for finite duration sequences) are essentially similar to those of the DFS for periodic sequences and result from the implied periodicity in the DFT representation.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

1.Signals and System
2.Fourier Transform
3.Z-Transform

Properties of Discrete Fourier Transform

- **Periodicity:**
  Due to the N-sample periodicity of the complex exponential basis functions ei2πnkN in the DFT and IDFT, the resulting transforms are also periodic with N samples.

  X(k+N)=X(k)x(n)=x(n+N)

- **Linearity :** $Ax(n) + Bx(n) \leftrightarrow AX(k) + BX(k)$
- **Time Shift:** $x(n-m) \leftrightarrow X(k)e^{-j2\pi km/N} = X(k)W_N^{k-m}$
- **Frequency Shift:**

$$x(n)e^{j2\pi km/N} \leftrightarrow X(k-m)$$

- **Time Reversal :** $x(n) \leftrightarrow X(-k)$
- **Circular convolution** $\sum_{m=0}^{N-1} x(m)y(n-m) = x(n)Oy(n) \leftrightarrow X(k)Y(k)$
- **Circular Convolution Property**

  Circular convolution is defined as x(n)*h(n)=N−1∑m=0x(m)x((n−m)modN)

Circular convolution of two discrete-time signals corresponds to multiplication of their DFTs: x(n)*h(n)=X(k)H(k)

- **Multiplication Property**

A similar property relates multiplication in time to circular convolution in frequency.x(n)h(n)=1NX(k)*H(k)

- **Multiplication:** $x(n)y(n) \leftrightarrow X(k)Y(k)$
- **Parseval's Theorem**

$$\sum_{n=0}^{N-1} |x(n)|^2 = N^{-1} \sum_{k=0}^{N-1} |X(k)|^2$$

Video Content / Details of website for further learning (if any):
http://fourier.eng.hmc.edu/e101/lectures/handout3/node7.html

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 409-414

Course Faculty

Verified by HOD

LECTURE HANDOUTS

L-4

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu.Prof./ECE

Unit : FOURIER ANALYSIS OF DISCRETE TIME SIGNALS Date of Lecture :

Topic of Lecture: Circular Convolution

Introduction :

- Find the response of filter with zero padding.The output samples $y(n)$ =Max(L,M)
- Properties of Convolution: 1- Commutative law, 2- Associative law , 3-Distributive law

Prerequisite knowledge for Complete understanding and learning of Topic:

1.Signals and System
2.Fourier Transform
3.Z-Transform

The Convolution Sum :

An arbitrary input signal *x(n)* in to a weighted sum of impulses, We are now ready to determine the response of any relaxed linear system to any Input signal. First, we denote the response y(n,k) of the system to the input unit Sample sequence at *n* = k by the special symbol *h(n, k), -∞<k*

$$y(n,k) \equiv h(n.k) = \mathcal{T}[\delta(n-k)]$$

< ∞. T h a t is,

if the input is the arbitrary signal x(n) that is expressed as a sum of weighted impulses, that is.

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k)$$

then the response of the system to *x(n)* is the corresponding sum of weighted outputs, that is,

Clearly, the above equation follows from the superposition property of linear systems, and is know n as the *superposition summation*.th en by the time-invariance property , the response of the system to the delayed unit sample sequence δ(n - *k)* is

$$h(n-k) = \mathcal{T}[\delta(n-k)]$$

Consequently , the *superposition summation* formula in reduces to

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

$$y(n) = T[x(n)] = T\left[\sum_{k=-\infty}^{\infty} x(k)\delta(n-k)\right]$$

$$= \sum_{k=-\infty}^{\infty} x(k)T[\delta(n-k)]$$

$$= \sum_{k=-\infty}^{\infty} x(k)h(n,k)$$

The above formula gives the response *y(n)* of the LTI system as a function of the input signal
x (n) and the unit sample (impulse) response *h(n)* is called a *convolution sum.*

To summarize, the process of computing the convolution between *x (k)* and *h(k)* involves the following four steps.

1. *Folding.* **Fold** *h(k)* **about** *k* **= 0 to obtain** *h (- k) .*

2. *Shifting,* Shift *h (– k)* by n_0 to the right (left) if n_0 is positive (negative), to obtain $h(n_0 - k)$.

3. *Multiplication.* Multiply *x (k)* by $h(n_0 - k)$ to obtain the product sequence $v_{n0}(k) = x(k)\,h(n_0 - k)$.

4. *Summation.* Sum all the values o f the product sequence $v_{n0}(k)$ to obtain the value of the output at time $n = n_0$.

This operation is analogous to linear convolution, but with a subtle difference
Consider two length-N sequences, g[n] and h[n], respectively

- Their linear convolution results in a length-(2N-1) sequence $y_L[n]$ given by In computing $y_L[n]$ we have assumed that both length-*N* sequences have been zero-padded to extend their lengths to 2N-1
- The longer form of $y_L[n]$ results from the time-reversal of the sequence *h*[*n*] and its linear shift to the right

The first nonzero value of $y_L[n]$ is $y_L[n]=g[0]h[0]$, and the last nonzero value is

$$y_L[2N-2]=g[N-1]h[N-1]$$

- To develop a convolution-like operation resulting in a length-*N* sequence $y_C[n]$, we need to define a circular time-reversal, and then apply a circular time-shift, resulting operation, called a circular convolution, is defined by

$$y_C[n] = \sum_{m=0}^{N-1} g[m]h[\langle n-m\rangle_N], \quad 0 \le n \le N-1$$

Since the operation defined involves two length-N sequences, it is often referred to as an N-point circular convolution, denoted as

$$y[n] = g[n] \circledN h[n]$$

The circular convolution is commutative, i.e.

$$g[n] \circledN h[n] = h[n] \circledN g[n]$$

Video Content / Details of website for further learning (if any):
http://www.wlxt.uestc.edu.cn/wlxt/ncourse/dsp/web/kj/Chapter3/3.6%20Circular%20Convolution.htm

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 415-420

Course Faculty

Verified by HOD

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu.Prof./ECE

Unit : FOURIER ANALYSIS OF DISCRETE TIME SIGNALS **Date of Lecture :**

Topic of Lecture: Filtering methods based on DFT

Introduction :

In signal processing, the function of a filter is to remove unwanted parts of the signal, such as random noise, or to extract useful parts of the signal, such as the components lying within a certain frequency range .

Prerequisite knowledge for Complete understanding and learning of Topic:

Filters may be classified as either digital or analog.

- Digital filters are implemented using a digital computer.
- Analog filters may be classified as either passive or active and are usually implemented with R, L, and C components and operational amplifiers.

Digital filter Specifications:

- These filters are unrealizable because (one of the following is sufficient)their impulse responses infinitely long non-causal.
- Their amplitude responses cannot be equal to a constant over a band of frequencies.
- The realizable squared amplitude response transfer function (and its differential) is continuous in Such functions
  - If IIR can be infinite at point but around that point cannot be zero.
  - If FIR cannot be infinite anywhere.

Detailed content of the Lecture:

DFT FOR LINEAR FILTERING:

Consider that input sequence x(n) of Length L & impulse response of same system is h(n) having M samples. Thus y(n) output of the system contains N samples where N=L+M-1. If DFT of y(n) also contains N samples then only it uniquely represents y(n) in time domain. Multiplication of two DFT s is equivalent to circular convolution of corresponding time domain sequences. But the length of x(n) & h(n) is less than N. Hence these sequences are appended with zeros to make their length N called as “Zero padding”.

METHOD 1: OVERLAP SAVE METHOD OF LINEAR FILTERING:

Step 1:

In this method L samples of the current segment and M-1 samples of the previous segment forms the input data block. Thus data block will be

X1(n) ={0,0,0,0,0,………………… ,x(0),x(1),…………….x(L-1)}

X2(n) ={x(L-M+1), …………….x(L-1),x(L),x(L+1),,,,,,,,,,,,x(2L-1)} X3(n) ={x(2L-M+1), …………….x(2L-1),x(2L),x(2L+2),,,,,,,,,,,,x(3L-1)}

Step2:

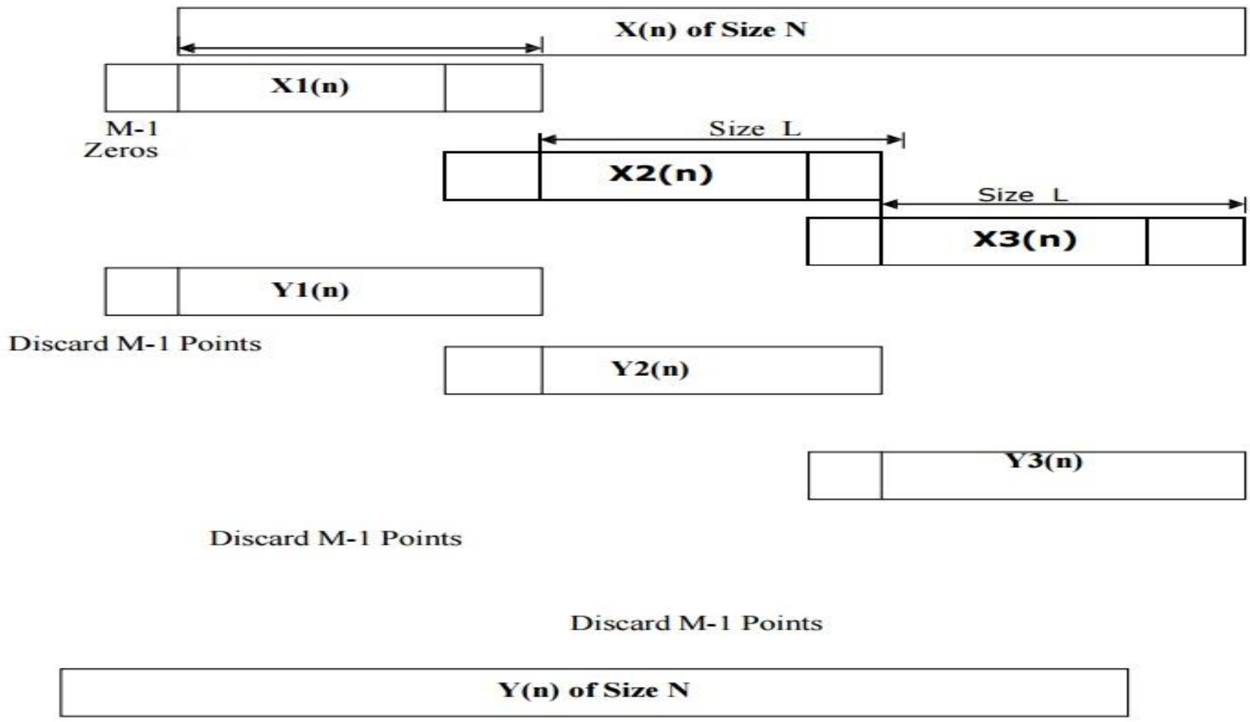
Unit sample response h(n) contains M samples hence its length is made N by padding zeros. Thus h(n) also contains N samples.

h(n)={ h(0), h(1), …………….h(M-1), 0,0,0,……………………(L-1 zeros)}

Step3: The N point DFT of h(n) is H(k) & DFT of m^{th} data block be xm(K) then corresponding DFT of output be Y`m(k)

Y`m(k)= H(k) xm(K)

Step 4: The sequence ym(n) can be obtained by taking N point IDFT of Y`m(k). Initial (M-1) samples in the corresponding data block must be discarded. The last L samples are the correct output samples. Such blocks are fitted one after another to get the final output.



METHOD 2: OVERLAP ADD METHOD OF LINEAR FILTERING:

Step 1: In this method L samples of the current segment and M-1 samples of the previous segment forms the input data block. Thus data block will be

X1(n) ={x(0),x(1),…………….x(L-1),0,0,0,……….}

X2(n) ={x(L),x(L+1),x(2L-1),0,0,0,0}

X3(n) ={x(2L),x(2L+2),,,,,,,,,,,,x(3L-1),0,0,0,0}

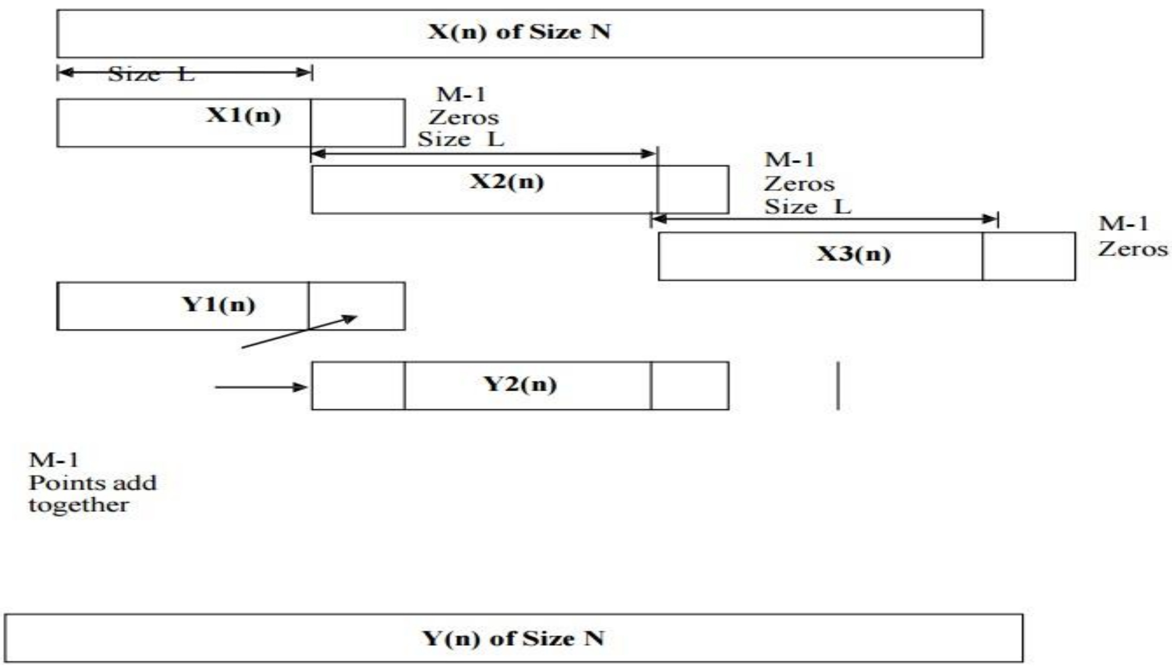
Step2: Unit sample response h(n) contains M samples hence its length is made N by padding zeros. Thus h(n) also contains N samples.

h(n)={ h(0), h(1), …………….h(M-1), 0,0,0,……………………(L-1 zeros)}

Step3: The N point DFT of h(n) is H(k) & DFT of m^{th} data block be xm(K) then corresponding DFT of output be Y`m(k)

Y`m(k)= H(k) xm(K)

Step 4: The sequence ym(n) can be obtained by taking N point IDFT of Y`m(k). Initial (M-1) samples are not discarded as there will be no aliasing. The last (M-1) samples of current output block must be added to the first M-1 samples of next output block. Such blocks are fitted one after another to get the final output.



The comparison between overlap save method and overlap add method are tabulated as follows,

S.No.	OVERLAP SAVE METHOD	OVERLAP ADD METHOD
1.	L samples of the current segment and (M-1) samples of the previous segment forms the input data block.	L samples from the input sequence and padding (M-1) zeros forms data block of size N
2.	Initial M-1 samples of output sequence are discarded which occurs due to aliasing effect.	There will be no aliasing in the output data block.
3.	To avoid loss of data due to aliasing last M-1 samples of each data record are saved.	Last M-1 samples of current output block must be added to the first M-1 samples of next output block. Hence called as overlap add method.

Video Content / Details of website for further learning (if any):

http://www.brainkart.com/article/Application-of-Discrete-Fourier-Transform(DFT)_13028/
https://www.youtube.com/watch?v=aEvBea7Mxaw

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 458-460

Course Faculty

Verified by HOD

LECTURE HANDOUTS

L-6

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu.Prof./ECE

Unit : FOURIER ANALYSIS OF DISCRETE TIME SIGNALS Date of Lecture :

Topic of Lecture: Fast Fourier Transform(FFT)

Introduction :

- A fast Fourier transform (FFT) is an algorithm that computes the discrete Fourier transform (DFT) of a sequence, or its inverse (IDFT).

Prerequisite knowledge for Complete understanding and learning of Topic:

- Discrete Fourier Transform (DFT) is a transform like Fourier transform used with digitized signals.
- It is the discrete version of the FT that views both the time domain and frequency domain as periodic.
- Fast Fourier Transform (FFT) is just an algorithm for fast and efficient computation of the DFT.
- The Fast Fourier Transform (FFT) is an approach to reduce the computational complexity that produces the same result as a DFT (same result, significantly fewer multiplications).
- The Fast Fourier Transform (FFT) is an implementation of the DFT which produces almost the same results as the DFT, but it is incredibly more efficient and much faster

Detailed content of the Lecture:

- FFT is an implementation of the DFT used for used for fast computation of the DFT.
- FFT is an efficient way of computing the DFT.
- The FFT is a fast algorithm for computing the DFT.
- To compute the DFT of an N-point sequence using equation (1) would take O(N2) multiplies and adds.
- The FFT algorithm computes the DFT using O(N log N) multiplies and adds.
- The inverse FFT (IFFT) is identical to the FFT, except one exchanges the roles of a and A, the signs of all the exponents of W are negated, and there's a division by N at the end.

FFT vs DFT Comparison Table

S.No.	FFT	DFT
1	FFT stands for Fast Fourier Transform	DFT stands for Discrete Fourier Transform
2	It is much faster version of DFT algorithm.	It is the discrete version of Fourier transform.
3	Various fast DFT computation techniques are collectively known as FFT algorithm.	It is the algorithm that transforms the time domain signals to the frequency domain components.
4	It is the implementation of DFT	It establishes the relation between the time domain and frequency domain representation
5	Applications include integer and polynomial multiplication, filtering algorithm, calculating Fourier series coefficients etc.	Applications of DFT includes solving partial differential applications, computing polynomial multiplication, spectral analysis etc.

<u>Algorithm of FFT and DFT:</u>

- The most commonly used FFT algorithm is the Cooley-Tukey algorithm.
- It's a divide and conquer algorithm for the machine calculation of complex Fourier series.
- It breaks the DFT into smaller DFTs.

- Other FFT algorithms include the Rader's algorithm, Wino grad Fourier transform algorithm, Chirp Z-transform algorithm, etc.

- The DFT algorithms can be either programmed on general purpose digital computers or implemented directly by special hardware.

- The FFT algorithm is used to compute the DFT of a sequence or its inverse.

- A DFT can be performed as O (N^2) in time complexity, whereas FFT reduces the time complexity in the order of O (NlogN).

Applications of FFT and DFT:

- FFT can be used in many digital processing systems across a variety of applications such as calculating a signal's frequency spectrum, solving partial differential applications, spectral analysis etc.

- FFT has been widely used for acoustic measurements in churches and concert halls.

- FFT include spectral analysis in analog video measurements, large integer and polynomial multiplication, filtering algorithms, computing isotopic distributions, calculating Fourier series coefficients, calculating convolutions, generating low frequency noise etc.

Video Content / Details of website for further learning (if any):

http://www.differencebetween.net/technology/difference-between-fft-and-dft/

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 415-417

Course Faculty

Verified by HOD

LECTURE HANDOUTS

L-7

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu.Prof./ECE**

Unit : **FOURIER ANALYSIS OF DISCRETE TIME SIGNALS** **Date of Lecture :**

Topic of Lecture: FFT Algorithms

Introduction :

- The Fast Fourier Transform (FFT) is a very efficient algorithm for performing a discrete Fourier transform.

Prerequisite knowledge for Complete understanding and learning of Topic:

FAST FOURIER ALGORITHM (FFT):

- Large number of the applications such as filtering, correlation analysis, and spectrum analysis require calculation of DFT.
- Direct computation of DFT requires large number of computations.
- Hence special algorithms are developed to compute DFT quickly called as Fast Fourier algorithms (FFT).
- The radix-2 FFT algorithms are based on divide and conquer approach.
- In this method, the N-point DFT is successively decomposed into smaller DFT s. Because of this decomposition, the numbers of computations are reduced.

Detailed content of the Lecture:

There are two types of FFT algorithms. They are,

- Decimation in Time
- Decimation in Frequency

RADIX-2 FFT ALGORITHMS:

DECIMATION IN TIME (DITFFT)

- There are three properties of twiddle factor WN

1) $W_N^{k+N} = W_N^K$ (Periodicity Property)

2) $W_N^{k+N/2} = -W_N^K$ (Symmetry Property)

3) $W_N^2 = W_{N/2}$.

N point sequence x(n) be splitted into two N/2 point data sequences f1(n) and f2(n).

f1(n) contains even numbered samples of x(n) and f2(n) contains odd numbered samples of x(n).

This splitted operation is called decimation.

Since it is done on time domain sequence it is called "**Decimation in Time**".

Thus

f1(m)=x(2m) where n=0,1,………….N/2-1

f2(m)=x(2m+1) where n=0,1,………….N/2-1

N point DFT is given as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn} \quad (1)$$

Since the sequence x(n) is splitted into even numbered and odd numbered samples, thus

$$X(k) = \sum_{m=0}^{N/2-1} x(2m) W_N^{2km} + \sum_{m=0}^{N/2-1} x(2m+1) W_N^{k(2m+1)} \quad (2)$$

$$X(k) = F1(k) + W_N^{k} F2(k) \quad (3)$$

$$X(k+N/2) = F1(k) - W_N^{k} F2(k) \text{ (Symmetry property)} \quad (4)$$

Fig 1 shows that 8-point DFT can be computed directly and hence no reduction in computation.

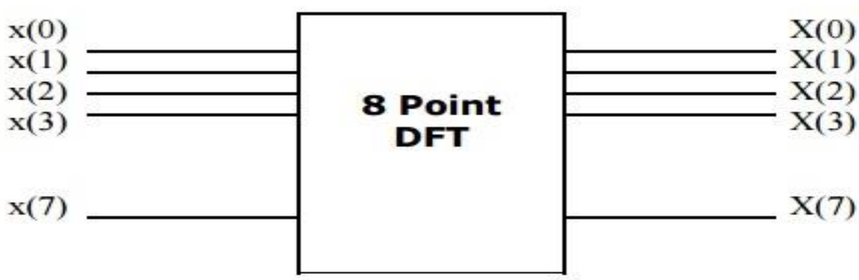


Fig 1. DIRECT COMPUTATION FOR N=8

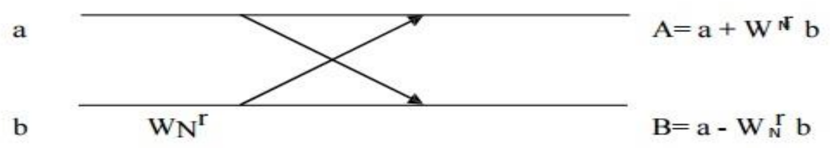


Fig 4. BUTTERFLY COMPUTATION (THIRD STAGE)

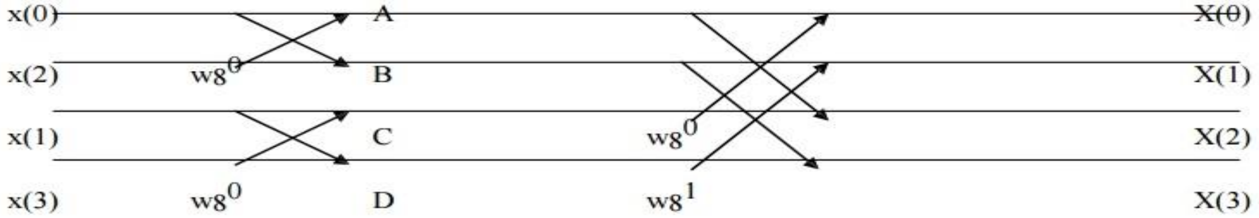


Fig 5. SIGNAL FLOW GRAPH FOR RADIX- DIT FFT N=4

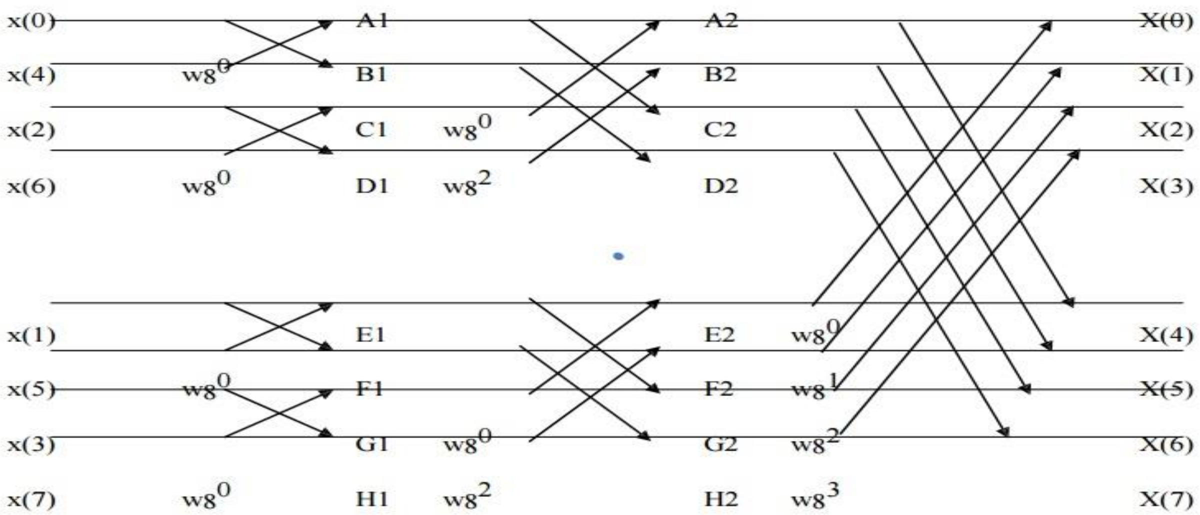


Fig 6. SIGNAL FLOW GRAPH FOR RADIX- DIT FFT N=8

Video Content / Details of website for further learning (if any):

https://www.slideshare.net/op205/fast-fourier-transform-presentation
https://www.slideshare.net/dhikadixiana/fast-fourier-transform-analysis

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 420-424

Course Faculty

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

LECTURE HANDOUTS

L-8

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu.Prof./ECE**

Unit : **FOURIER ANALYSIS OF DISCRETE TIME SIGNALS** **Date of Lecture :**

Topic of Lecture: Decimation in time Algorithms

Introduction :

- Decimation in time FFT algorithms are based upon decomposition of the input sequence into smaller and smaller sub sequences.
- DIT FFT input sequence is in bit reversed order while the output sequence is in natural order.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Decimation is the process of reducing the sampling rate.
- Decimation is a term that historically means the removal of every tenth one.
- But in signal processing, decimation by a factor of 10 actually means keeping only every tenth sample.
- This factor multiplies the sampling interval or, equivalently, divides the sampling rate.

Detailed content of the Lecture:

Decimation in Time:

The DFT is defined by,

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \quad k = 0, 1, 2, \ldots, N-1,$$

where $x(n)$ is the input signal amplitude at time n , and

$$W_N \triangleq e^{-j\frac{2\pi}{N}}. \quad \text{(primitive } N\text{th root of unity)}$$

Note that $W_N^N = 1$

When N is even, the DFT summation can be split into sums over the odd and even indexes of the input signal:

$$X(\omega_k) \triangleq \text{DFT}_{N,k}\{x\} \triangleq \sum_{n=0}^{N-1} x(n) e^{-j\omega_k nT}, \quad \omega_k \triangleq \frac{2\pi k}{NT}$$

$$= \sum_{\substack{n=0 \\ n \text{ even}}}^{N-2} x(n) e^{-j\omega_k nT} + \sum_{\substack{n=0 \\ n \text{ odd}}}^{N-1} x(n) e^{-j\omega_k nT}$$

$$= \sum_{n=0}^{\frac{N}{2}-1} x(2n) e^{-j2\pi \frac{k}{N/2} n} + e^{-j2\pi \frac{k}{N}} \sum_{n=0}^{\frac{N}{2}-1} x(2n+1) e^{-j2\pi \frac{k}{N/2} n},$$

$$= \sum_{n=0}^{\frac{N}{2}-1} x_e(n) W_{N/2}^{kn} + W_N^k \sum_{n=0}^{\frac{N}{2}-1} x_o(n) W_{N/2}^{kn}$$

$$\triangleq \text{DFT}_{\frac{N}{2},k}\{\text{DOWNSAMPLE}_2(x)\}$$

$$+ W_N^k \cdot \text{DFT}_{\frac{N}{2},k}\{\text{DOWNSAMPLE}_2[\text{SHIFT}_1(x)]\},$$

where $x_e(n) \triangleq x(2n)$ and $x_o(n) \triangleq x(2n+1)$ denote the even- and odd-indexed samples from x.

Thus, the length N DFT is computable using two length $N/2$ DFTs. The complex factors $W_N^k = e^{-j\omega_k} = \exp(-j2\pi k/N)$ are called twiddle factors.

The splitting into sums over even and odd time indexes is called decimation in time.

Video Content / Details of website for further learning (if any):
https://www.dsprelated.com/freebooks/mdft/Decimation_Time.html

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 429-431

Course Faculty

Verified by HOD

LECTURE HANDOUTS **L-9**

ECE **II/IV**

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu.Prof./ECE

Unit : FOURIER ANALYSIS OF DISCRETE TIME SIGNALS Date of Lecture :

Topic of Lecture: Decimation in frequency algorithms

Introduction :

- The Decimation in Frequency FFT algorithms are based upon decomposition of the output sequence into smaller and smaller sub sequences.
- In Decimation in Frequency FFT, input sequence is in natural order.
- DFT should be read in bit reversed order.

Prerequisite knowledge for Complete understanding and learning of Topic:

- The decimation-in-frequency FFT is a flow-graph reversal of the decimation-in-time FFT: it has the same twiddle factors (in reverse pattern) and the same operation counts.

Detailed content of the Lecture:

- In a decimation-in-frequency radix-2 FFT as illustrated in Figure, the output is in bit-reversed order.

In DIF N Point DFT is splitted into N/2 points DFT s. X(k) is splitted with k even and k odd this is called Decimation in frequency(DIF FFT).

N point DFT is given as,

$$X(k) = \sum_{n=0}^{N-1} x(n)\, W_N^{kn} \qquad (1)$$

Since the sequence x(n) is splitted N/2 point samples, thus

$$X(k) = \sum_{m=0}^{N/2-1} x(n)\, W_N + \sum_{m=0}^{N/2-1} x(n+N/2)\, W_N \quad (2)$$

$$X(k) = \sum_{m=0}^{N/2-1} x(n)\, W_N^{kn} + W_N^{kN/2} \sum_{m=0}^{N/2-1} x(n+N/2)\, W_N^{kn}$$

$$X(k) = \sum_{m=0}^{N/2-1} x(n)\, W_N^{kn} + (-1)^k \sum_{m=0}^{N/2-1} x(n+N/2)\, W_N^{kn}$$

$$X(k) = \sum_{m=0}^{N/2-1} \left[x(n) + (-1)^k\, x(n+N/2) \right] W_N^{kn} \quad (3)$$

Let us split X(k) into even and odd numbered samples

$$X(2k) = \sum_{m=0}^{N/2-1} \left[x(n) + (-1)^{2k}\, x(n+N/2)\, W_N^{2kn} \right] \quad (4)$$

$$X(2k+1) = \sum_{m=0}^{N/2-1} \left[x(n) + (-1)^{(2k+1)}\, x(n+N/2)\, W_N^{(2k+1)n} \right] \quad (5)$$

Equation (4) and (5) are thus simplified as

$$g1(n) = x(n) + x(n+N/2)$$
$$g2(n) = x(n) - x(n+N/2)\, W_N^{n}$$

Fig 1 shows **Butterfly computation** in DIF FFT.

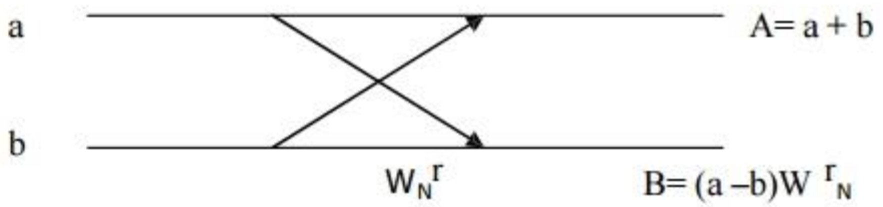


Fig 1. BUTTERFLY COMPUTATION

Fig 2 shows signal flow graph and stages for computation of radix-2 DIF FFT algorithm of N=4

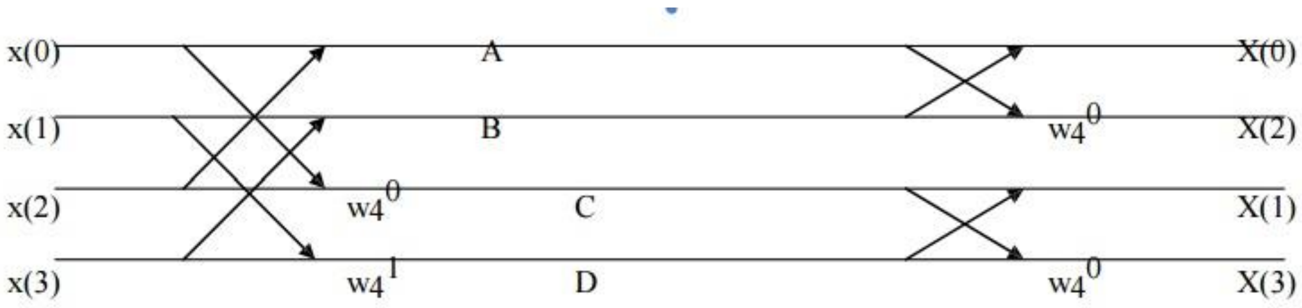


Fig 2. SIGNAL FLOW GRAPH FOR RADIX- DIF FFT N=4

DIFFERENCE BETWEEN DECIMATION IN TIME FFT AND DECIMATION IN FREQUENCY FFT:

S.No.	DIT FFT	DIF FFT
1	DIT FFT algorithms are based upon decomposition of the input sequence into smaller and smaller sub sequence	DIF FFT algorithms are based upon decomposition of the output sequence into smaller and smaller sub sequence.
2	In this input sequence x(n) is splitted into even and odd numbered samples	In this input sequence X(k) is considered to be splitted into even and odd numbered samples
3	Splitting operation is done on time domain sequence.	Splitting operation is done on frequency domain sequence.
4	In DIT FFT input sequence is in bit reversed order while the output sequence is in natural order	In DIF FFT, input sequence is in natural order and DFT should be read in bit reversed order.

Video Content / Details of website for further learning (if any):

http://www.brainkart.com/article/Decimation-In-Frequency-(DIFFFT)_13033/

Important Books/Journals for further learning including the page nos.:

John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 434-441

Course Faculty

Verified by HOD

LECTURE HANDOUTS

L-10

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANESH BABU.Prof./ECE

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: Structure of IIR

Introduction :

- IIR system has infinite duration unit sample response h(n)=0 for n<0
- Thus the unit sample response exists for the duration from 0 to ∞
- IIR filter is usually more efficient design in terms of computation time and memory requirement.
- IIR systems usually requires less processing time and storage as compared with FIR

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

1.Signals and System
2.Fourier Transform
3.Z-Transform

Introduction

consider different IIR system s structures described by the difference equation given by the system function. Just as in the case o f FIR system s, there are several types o f structures or realizations, including direct-form structures, cascade-form structures, lattice structures, and lattice-ladder structures. In addition, IIR systems lend themselves to a parallel form realization. We begin by describing two direct-form realizations.

DIRECT FORM STRUCTURES:

The rational system function as given by that characterizes an IIR system can be viewed as two systems in cascade, that is,

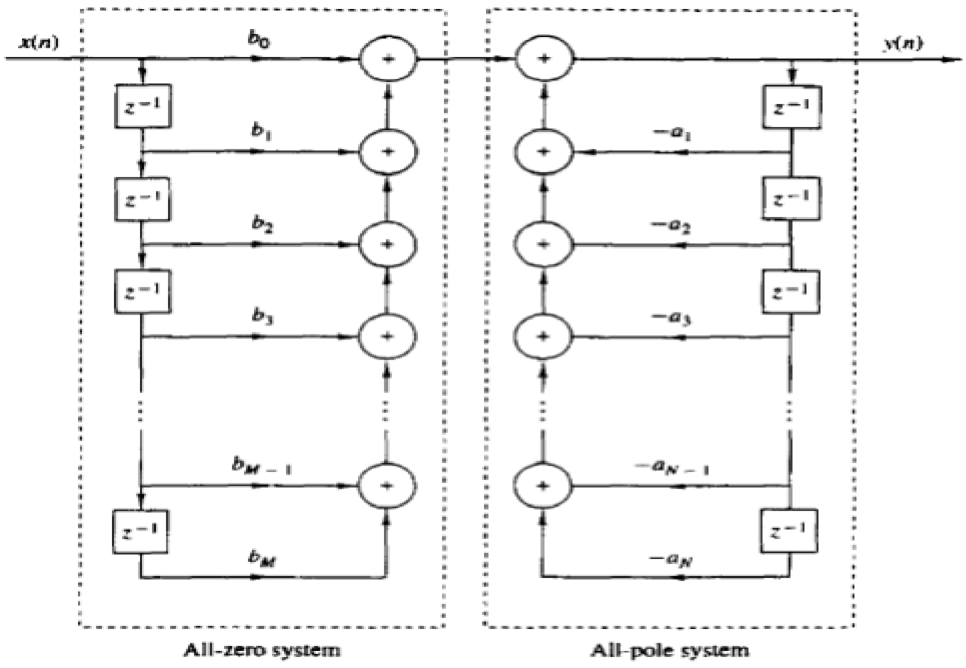
$$H(z) = H_1(z)H_2(z) \qquad 1$$

where $H_1(z)$ consists of the zeros of $H(z)$, and $H_2(z)$ consists of the poles of $H(z)$,

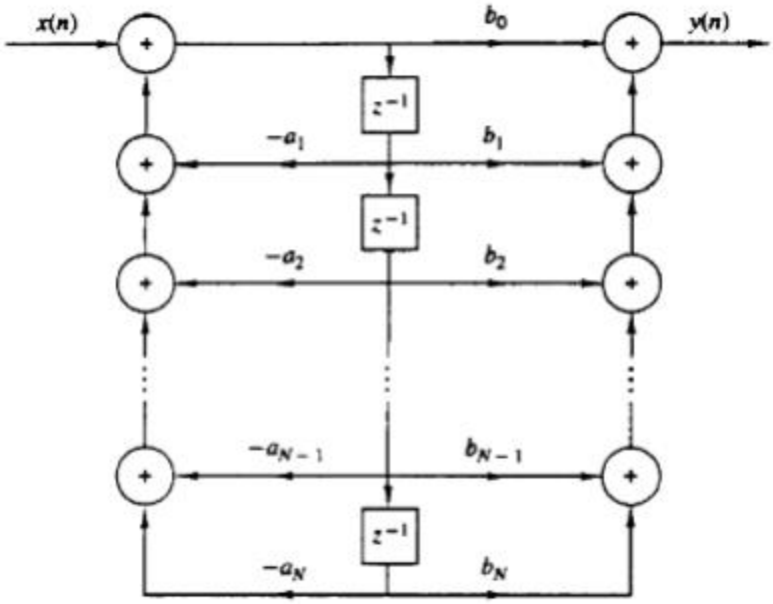
$$H_1(z) = \sum_{k=0}^{M} b_k z^{-k} \qquad 2$$

and

$$H_2(z) = \frac{1}{1 + \sum_{k=1}^{N} a_k z^{-k}} \qquad 3$$



DIRECT FORM II



Cascade-Form Structures

Let us consider a high-order IIR system with system function given by equation. Without loss o f generality we assume that $N > M$. T h e system can be factored into a cascade o f second-order subsystem s, such that $H(z)$ can b e expressed as

$$H(z) = \prod_{k=1}^{K} H_k(z)$$

where K is the integer part of $(N+1)/2$. $H_k(z)$ has the general form

$$H_k(z) = \frac{b_{k0} + b_{k1}z^{-1} + b_{k2}z^{-2}}{1 + a_{k1}z^{-1} + a_{k2}z^{-2}}$$

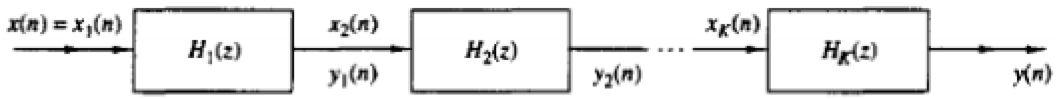
The general form of the cascade structure is illustrated in Fig. 7.19. If we use the direct form II structure for each of the subsystems, the computational algorithm for realizing the IIR system with system function $H(z)$ is described by the following set of equations.

$$y_0(n) = x(n) \tag{7.3.16}$$

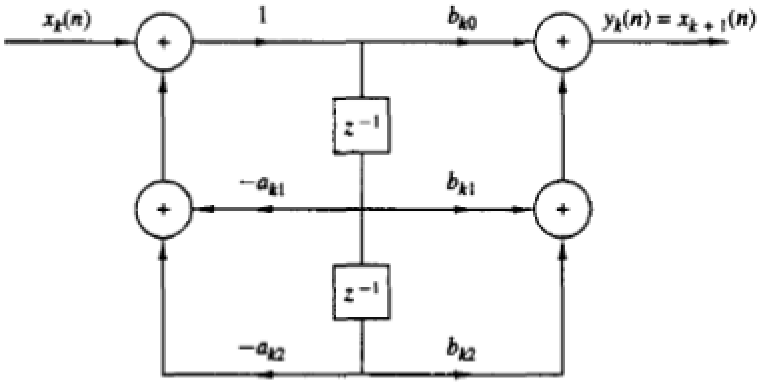
$$w_k(n) = -a_{k1}w_k(n-1) - a_{k2}w_k(n-2) + y_{k-1}(n) \qquad k = 1, 2, \ldots, K \tag{7.3.17}$$

$$y_k(n) = b_{k0}w_k(n) + b_{k1}w_k(n-1) + b_{k2}w_k(n-2) \qquad k = 1, 2, \ldots, K \tag{7.3.18}$$

$$y(n) = y_K(n) \tag{7.3.19}$$



(a)

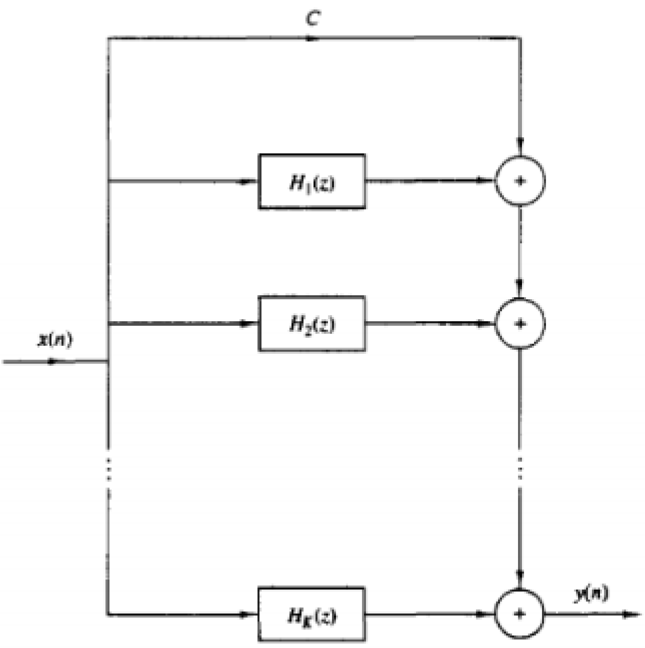


(b)

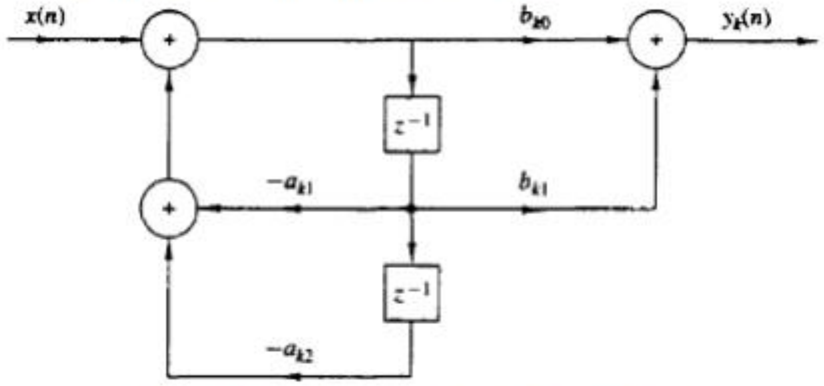
Parallel-Form Structures

A parallel-form realization o f an IIR system can be obtained by performing a partial-fraction expansion o f $H(z)$. Without loss o f generality, w e again assume that $N > M$ and that the poles are distinct. Then, by performing a partial-fraction expansion o f $H(z)$, we obtain the result

$$H(z) = C + \sum_{k=1}^{N} \frac{A_k}{1 - p_k z^{-1}}$$



The realization of second order form is given by



The general form of parallel form of structure is f\given by

$$w_k(n) = -a_{k1}w_k(n-1) - a_{k2}w_k(n-2) + x(n) \qquad k = 1, 2, \ldots, K$$

$$y_k(n) = b_{k0}w_k(n) + b_{k1}w_k(n-1) \qquad k = 1, 2, \ldots, K$$

$$y(n) = Cx(n) + \sum_{k=1}^{K} y_k(n)$$

Video Content / Details of website for further learning (if any):
https://aits-tpt.edu.in/wp-content/uploads/2018/08/Structures-of-FIR-and-IIR-systems.pdf

Important Books/Journals for further learning including the page nos.:
John G. Proakis & Dimitris G.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg:298-305

CourseFaculty

Verified by HOD

LECTURE HANDOUTS

L-11

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANESH BABU. Prof./ECE

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: Analog Filter Design

Introduction : (Maximum 5 sentences)

- Analog filters are designed with various components like resister, inductor and capacitance
- Analog filters less accurate & because of component tolerance of active components & more sensitive to environmental chandes
- An analog filter can only be changed by redesigning the filter circuit

Prerequisite knowledge for Complete understanding and learning of Topic:
(Max. Four important topics)

1.Signals and System
2.Filter
3.Analog Signal Processing

Detailed content of the Lecture:

Digital filters can be designed using analog design methods by following these steps:

1. Filter specifications are specified in the digital domain. The filter type (highpass, lowpass,bandpass etc.) is specified.
2. An equivalent lowpass filter is designed that meets these specifications.
3. The analog lowpass filter is transformed using spectral transformations into the correct type of filter.
4. The analog filter is transformed into a digital filter using a particular mapping.
5. There are many different types of spectral transformations and there are many mappings from analog to digital filters. the most famous mapping is known as the bilinear transform, and we will discuss that in a different chapter.

SELECTION OF THE FILTER TYPE

The selection of the digital filter type *i.e.*, whether an IIR and FIR digital filter to be employed ; depends on the nature of the problem and on the specification of the desired frequency response. For example, FIR filters are used in filtering problems where there is a requirement for a linear phase characteristic within the passband of the filter. When linear phase is not a requirement, either an IIR or FIR filter can be used. However, in most cases, the order (N_{FIR}) of an FIR filter is considerably higher than the order (N_{IIR}) of an equivalent IIR filter meeting the same magnitude specifications. It has been shown that for most practical filter specifications, the ratio N_{FIR}/N_{IIR} is typically of order of ten or more and as a result, IIR filter is usually computationally more efficient.

In this chapter we shall discuss techniques for designing IIR filters from the analog filters, with the restriction that the filters be realizable and, of course, stable. There are four different methods which are available under IIR filter design, these are,

1. Impulse invariance method
2. Bilinear transformation method
3. Matched *z*-transform technique
4. Approximation of derivatives.

We shall concentrate only the first two methods.

IIR Filter Design by Impulse Invariance

A technique for digitizing an analog filter is called impulse invariance transformation. The objective of this method is to develop an IIR filter transfer function whose impulse response is the sampled version of the impulse response of the analog filter. The main idea behind this technique is to preserve the frequency response characteristics of the analog filter. In the consequence of the result, the frequency response of the digital filter is an aliased version of the frequency response of the corresponding analog filter.

To develop the necessary design formula for impulse invariance method, consider a causal and stable "analog" transfer function $H_a(s)$. Its impulse response $h_a(t)$ is given by inverse Laplace transform of $H_a(s)$, *i.e.*,

$$h_a(t) = L^{-1}\{H_a(s)\} \qquad \ldots(1)$$

In this method, we require that unit sample response $h(n)$ of the desired causal digital transfer function H(*z*) be given by the sampled version of $h_a(t)$ sampled at uniform interval of T seconds.

i.e.,
$$h(n) = h_a(nT) \quad n = 0, 1, 2 \qquad \ldots(2)$$

where T is the sampling period

To investigate the mapping of points between *z*-plane and *s*-plane implied by the sampling process, the *z*-transform is related to the Laplace transform of $h_a(t)$ as

$$H(z)\big|_{z=e^{sT}} = Z\{h(n)\} = Z\{h_a(nT)\} \qquad \ldots(3)$$

$$= \frac{1}{T}\sum_{k=-\infty}^{\infty} H_a\left(s + j\frac{2\pi k}{T}\right) \qquad \ldots(4)$$

where,
$$H(z) = \sum_{n=0}^{\infty} h(n)\, z^{-n} \qquad \ldots(5)$$

and
$$H(z)\big|_{z=e^{sT}} = \sum_{n=0}^{\infty} h(n) e^{-sTn} \qquad \ldots(6)$$

Video Content / Details of website for further learning (if any):
https://en.wikibooks.org/wiki/Digital_Signal_Processing/Analog_Filter_Design

Important Books/Journals for further learning including the page nos.:
John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg:262 -268

CourseFaculty

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

LECTURE HANDOUTS

L-12

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANESH BABU. Prof./ECE

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: Discrete time IIR filter from analog filter

Introduction :

- To design a discrete-time lowpass filter using both the impulse invariance method and the bilinear transform method.

Prerequisite knowledge for Complete understanding and learning of Topic: (Max. Four important topics)

1.Infinite Impulse Response

Properties of Discrete Fourier Transform

IIR FILTER DESIGN BY THE BILINEAR TRANSFORMATION

The IIR filter design using (i) approximation of derivatives method and (ii) the impulse invariant method are appropriate for the design of low-pass filters and bandpass filters whose resonant frequencies are low. These techniques are not suitable for high-pass or band-reject filters. This limitation is overcome in the mapping technique called the ***bilinear transformation***. This transformation is a one-to-one mapping from the s-domain to the z-domain. That is, the bilinear transformation is a conformal mapping that transforms the $j\Omega$-axis into the unit circle in the z-plane only once, thus avoiding aliasing of frequency components. Also, the transformation of a stable analog filter results in a stable digital filter as all the poles in the left half of the s-plane are mapped onto points inside the unit circle of the z-domain. The bilinear transformation is obtained by using the trapezoidal formula for numerical integration. Let the system function of the analog filter be

$$H(s) = \frac{b}{s+a}$$

The differential equation describing the analog filter can be obtained rom Eq. 2 as shown below.

$$H(s) = \frac{Y(s)}{X(s)} = \frac{b}{s+a}$$

$$sY(s) + aY(s) = bX(s)$$

'aking inverse Laplace transform,

$$\frac{dy(t)}{dt} + ay(t) = bx(t)$$

$$\int_{nT-T}^{nT} \frac{dy(t)}{dt}\,dt + a\int_{nT-T}^{nT} y(t)\,dt = b\int_{nT-T}^{nT} x(t)\,dt$$

The trapezoidal rule for numeric integration is given by

$$\int_{nT-T}^{nT} a(t)\,dt = \frac{T}{2}\,[a(nT) + a(nT-T)]$$

Applying Eq. 5 in Eq. 4 we get

$$y(nT) - y(nT-T) + \frac{aT}{2}y(nT) + \frac{aT}{2}y(nT-T) = \frac{bT}{2}x(nT) + \frac{bT}{2}x(nT-T)$$

Taking z-transform, the system function of the digital filter is,

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b}{\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right) + a}$$

Comparing Eqs. 2 and 6 we get,

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

Video Content / Details of website for further learning (if any):
https://eeweb.engineering.nyu.edu/iselesni/EL713/iir/iir.pdf

Important Books/Journals for further learning including the page nos.:
John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 298-305

CourseFaculty

Verified by HOD

LECTURE HANDOUTS | **L-13**

ECE | **II/IV**

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANESH BABU.Prof./ECE

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: IIR filter design by Impulse Invariance

Introduction :

- Impulse invariance is a technique for designing discrete-time infinite-impulse-response (IIR) to produce the impulse response of the discrete-time system.
- The frequency response of the discrete-time system will be a sum of shifted copies of the frequency response of the continuous-time system

Prerequisite knowledge for Complete understanding and learning of Topic:

1. Infinite Impulse Response
2.Filter

Detailed content of the Lecture:

IIR Filter Design by Impulse Invariance

A technique for digitizing an analog filter is called impulse invariance transformation. The objective of this method is to develop an IIR filter transfer function whose impulse response is the sampled version of the impulse response of the analog filter. The main idea behind this technique is to preserve the frequency response characteristics of the analog filter. In the consequence of the result, the frequency response of the digital filter is an aliased version of the frequency response of the corresponding analog filter.

To develop the necessary design formula for impulse invariance method, consider a causal and stable "analog" transfer function $H_a(s)$. Its impulse response $h_a(t)$ is given by inverse Laplace transform of $H_a(s)$, *i.e.,*

$$h_a(t) = L^{-1}\{H_a(s)\} \qquad ...(1)$$

In this method, we require that unit sample response $h(n)$ of the desired causal digital transfer function H(z) be given by the sampled version of $h_a(t)$ sampled at uniform interval of T seconds.

i.e.,
$$h(n) = h_a(nT) \quad n = 0, 1, 2 \qquad ...(2)$$

where T is the sampling period

To investigate the mapping of points between z-plane and s-plane implied by the sampling process, the z-transform is related to the Laplace transform of $h_a(t)$ as

$$H(z)\big|_{z=e^{sT}} = Z\{h(n)\} = Z\{h_a(nT)\} \qquad ...(3)$$

$$= \frac{1}{T}\sum_{k=-\infty}^{\infty} H_a\left(s + j\frac{2\pi k}{T}\right) \qquad ...(4)$$

where,
$$H(z) = \sum h(n)\, z^{-n} \qquad ...(5)$$

wnere $s = \sigma + j\Omega$

Let us examine the transform $z = e^{ST}$ of eqn. (4) which can be written alternatively as,

$$z = e^{sT}$$

For $s = \sigma + j\Omega$

$$z = \gamma e^{j\omega} = e^{sT} = e^{\sigma T} = e^{j\Omega T}.$$

This then implies $\Omega = e^{\sigma T}$

$$\omega = \Omega T$$

where Ω is analog frequency and

ω is frequency in digital domain.

Video Content / Details of website for further learning (if any):
https://flylib.com/books/en/2.729.1/impulse_invariance_iir_filter_design_method.html

Important Books/Journals for further learning including the page nos.:
John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 415-420

CourseFaculty

Verified by HOD

LECTURE HANDOUTS

L-14

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANESH BABU.Prof./ECE

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: IIR filter design by Bilinear Transformation

Introduction :

- Impulse invariance is a technique for designing discrete-time infinite-impulse-response (IIR) to produce the impulse response of the discrete-time system.
- The frequency response of the discrete-time system will be a sum of shifted copies of the frequency response of the continuous-time system

Prerequisite knowledge for Complete understanding and learning of Topic:

1. Infinite Impulse Response
2. Filter

Detailed content of the Lecture:

The bilinear transform (also known as Tustin's method) is used in digital signal processing and discrete-time control theory to transform continuous-time system representations to discrete-time and vice versa.The bilinear transform is a special case of a conformal mapping (namely, a Möbius transformation), often used to convert a transfer function of a linear, time-invariant (LTI) filter in the continuous-time domain (often called an analog filter) to a transfer function of a linear, shift-invariant filter in the discrete-time domain (often called a digital filter although there are analog filters constructed with switched capacitors that are discrete-time filters). It maps positions on the in the s-plane to the unit circle in the z-plane. Other bilinear transforms can be used to warp the frequency response of any discrete-time linear system (for example to approximate the non-linear frequency resolution of the human auditory system) and are implementable in the discrete domain by replacing a system's unit delays {\displaystyle \left(z^{-1}\right)} with first order all-pass filters.

The transform preserves stability and maps every point of the frequency response of the continuous-time filter, to a corresponding point in the frequency response of the discrete-time filter, although to a somewhat different frequency, as shown in the Frequency warping section below. This means that for every feature that one sees in the frequency response of the analog filter, there is a corresponding feature, with identical gain and phase shift, in the frequency response of the digital filter but, perhaps, at a somewhat different frequency. This is barely noticeable at low frequencies but is quite evident at frequencies close to the Nyquist frequency.

Video Content / Details of website for further learning (if any): https://flylib.com/books/en/2.729.1/impulse_invariance_iir_filter_design_method.html
Important Books/Journals for further learning including the page nos.: John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 415-420

CourseFaculty

Verified by HOD

LECTURE HANDOUTS

L-15

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANESH BABU.Prof./ECE

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: Bilinear transformation

Introduction :

The Bilinear transform is a mathematical relationship which can be used to convert the transfer function of a particular filter in the complex Laplace domain into the z-domain, and vice-versa. The resulting filter will have the same characteristics of the original filter, but can be implemented using different techniques. The Laplace Domain is better suited for designing analog filter components, while the Z-Transform is better suited for designing digital filter components.

Prerequisite knowledge for Complete understanding and learning of Topic:

1.Infinite Impulse Response

Detailed content of the Lecture:

Bilinear transformation technique:

This is the most common method for transforming the system function $H_a(s)$ of an analogue filter to the system function H(z) of an IIR discrete time filter. It is not the only possible transformation, but a very useful and reliable one.

Introduction

Definition: Given analogue transfer function $H_a(s)$, replace s by :

$$\frac{2}{T}\left[\frac{z-1}{z+1}\right]$$

to obtain H(z). For convenience we can take T=1.

Problem *Obtain H(z) from $H_a(s)$ when T = 1 sec and* $H_a(s) = \frac{s^3}{(s+1)(s^2+s+1)}$.

Sol. Given that, $H_a(s) = \frac{s^3}{(s+1)(s^2+s+1)}$.

Put $s = \frac{2}{T}\frac{1-z^{-1}}{1+z^{-1}}$ in $H_a(s)$ to get H(z).

$$H(z) = \frac{\left[\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)\right]^3}{\left[\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)+1\right]\left[\left(\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)\right)^2+\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)+1\right]}$$

$$= \frac{8(1-z^{-1})^3}{\left[2\left(1-z^{-1}\right)+T(1+z^{-1})\right]\left[4(1-z^{-1})^2+2T(1-z^{-1})(1+z^{-1})+T^2(1+z^{-1})^2\right]}$$

But T = 1 sec.

$$H(z) = \frac{8(1-z^{-1})^3}{(3-z^{-1})(7-6z^{-1}+3z^{-2})}$$

$$H(z) = \frac{2.67(z^{-1}-1)^3}{(z^{-2}-2z^{-1}+2.33)(z^{-1}-3)}.$$

Problem *Design a digital Butterworth filter satisfying the constraints*

$$0.707 \le |H(e^{j\omega})| \le 1 \quad for\ 0 \le \omega \le \frac{\pi}{2}$$

$$|H(e^{j\omega})| \le 0.2 \quad for \quad \frac{3\pi}{4} \le \omega \le \pi.$$

with T = 1 sec using The bilinear transformation

Sol. Bilinear transformation

Given that $\frac{1}{\sqrt{1+\varepsilon^2}} = 0.707$; $\frac{1}{\sqrt{1+\lambda^2}} = 0.2$, $\omega_p = \frac{\pi}{2}$; $\omega_s = \frac{3\pi}{4}$

The analog frequency ratio is

$$\frac{\Omega_s}{\Omega_p} = \frac{\frac{2}{T}\tan\frac{\omega_s}{2}}{\frac{2}{T}\tan\frac{\omega_p}{2}} = \frac{\tan\frac{3\pi}{8}}{\tan\frac{\pi}{4}} = 2.414$$

The order of the filter,

$$N \ge \frac{\log \lambda/\varepsilon}{\log\frac{\Omega_s}{\Omega_p}}.$$

From the given data λ = 4.898, ε = 1,

So, $$N \ge \frac{\log 4.898}{\log 2.414} = 1.803.$$

Rounding N to nearest higher value we get N = 2.

We know $$\Omega_c = \frac{\Omega_p}{(\varepsilon)^{1/N}} = \Omega_p \qquad (\because \ \varepsilon = 1)$$

$$= \frac{2}{T}\tan\frac{\omega_p}{2} = 2\tan\frac{\pi}{4} = 2 \text{ rad/sec}.$$

The transfer function of second order normalised Butterworth filter is,

$$H(s) = \frac{1}{s^2+\sqrt{2}s+1}.$$

$H_a(s)$ for $\Omega_c = 2$ rad/sec can be obtained by substituting

$s \rightarrow s/2$ in $H(s)$

i.e.,
$$H_a(s) = \frac{1}{(s/2)^2 + \sqrt{2}(s/2) + 1} = \frac{4}{s^2 + 2.828s + 4}.$$

By using bilinear transformation H(z) can be obtained as

$$H(z) = H(s)\Big|_{s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)}.$$

Thus
$$H(z) = \frac{4}{s^2 + 2.828s + 4}\Bigg|_{s = 2\left(\frac{1-z^{-1}}{1+z^{-1}}\right)} \qquad (\because \; T = 1 \text{ sec})$$

$$= \frac{4(1+z^{-1})^2}{4(1-z^{-1})^2 + 2.828(1-z^{-2}) + 4(1+z^{-1})^2}$$

$$= \frac{0.2929(1+z^{-1})^2}{1+0.1716z^{-2}}.$$

Video Content / Details of website for further learning (if any):
https://en.wikibooks.org/wiki/Digital_Signal_Processing/Bilinear_Transform#The_Bilinear_Transform

Important Books/Journals for further learning including the page nos.:
John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 285-290

CourseFaculty

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-16

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANES BABU.Prof./ECE

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: Problems in IIR filter

Introduction :

The impulse invariance method of IIR filter design is based upon the notion that we can design a discrete filter whose time-domain impulse response is a sampled version of the impulse response of a continuous analog filter

Prerequisite knowledge for Complete understanding and learning of Topic:

1.Infinite Impulse Response

Detailed content of the Lecture:

To explore the effect of the impulse invariance design method on the characteristics of resultant filter, let us consider the system function of the analog filter in the partial fraction form. Assume that the poles of the analog filter are distinct *i.e.*,

$$H_a(s) = \sum_{k=1}^{N} \frac{A_k}{s - p_k} \quad \ldots(7)$$

where $\{A_k\}$ are the co-efficients in the partial fraction expansion and

p_k are the poles of the analog filter. The impulse response $h_a(t)$ corresponding to eqn. (7) has the form

$$h_a(t) = \sum_{k=1}^{N} A_k e^{p_k t} u_a(t) \quad \ldots(8)$$

where $u_a(t)$ is the continuous time step function.

If we sample $h_a(t)$ periodically at $t = nT$, we have

$$h(n) = h_a(nT) = \sum_{k=1}^{N} A_k e^{p_k nT} u_a(nT) \quad \ldots(9)$$

Now, the system function H(z) of the digital filter is the z-transform of this sequence and is defined by

$$H(z) = z\{h(n)\}.$$

$$H(z) = \sum_{n=0}^{\infty} h(n) z^{-n} \quad ...(10)$$

Using eqn. (10) the system function becomes

$$H(z) = \sum_{n=0}^{\infty} \sum_{k=1}^{N} A_k e^{p_k nT} z^{-n} \quad ...(11)$$

$$= \sum_{k=1}^{N} \sum_{k=0}^{\infty} (e^{p_k T} z^{-1})^n \quad ...(12)$$

$$H(z) = \sum_{k=1}^{N} A_k \cdot \frac{1}{1 - e^{p_k T} z^{-1}} \quad ...(13)$$

provided that $|e^{p_k T}| < 1$, which is always satisfied if $p_k < 0$, indicating that $H_a(s)$ is a stable transfer function. From the eqn. (13) we observe that the digital filter has poles at

$$z_k = e^{p_k T} \quad k = 1, 2, \ldots\ldots\ldots N$$

Comparing the expression (13) and (7), we see that the impulse invariance transformation is accomplished by the mapping.

$$\frac{1}{s - p_k} \longrightarrow \frac{1}{1 - e^{p_k T} z^{-1}} \quad ...(14)$$

$$\frac{1}{s + p_k} \longrightarrow \frac{1}{1 - e^{-p_k T} z^{-1}}.$$

Problem 1. *For the analog transfer function* $H_a(s) = \frac{2}{(s+1)(s+2)}$ *determine the H(z) using impulse invariance method.*

Sol.
$$H_a(s) = \frac{2}{(s+1)(s+2)} = \frac{2}{s+1} - \frac{2}{s+2}$$

Using the impulse invariance transformation of eqn. (14), the digital filter transfer function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{2}{1 - e^{-T} z^{-1}} - \frac{2}{1 - e^{-2T} z^{-1}} = \frac{2e^{-T}(1 - e^{-T}) z^{-1}}{(1 - e^{-T} z^{-1})(1 - e^{-2T} z^{-1})}.$$

Problem 2. *Convert the analog filter with system function*

$$H_a(s) = \frac{s+2}{(s+1)(s+3)}$$

into the digital IIR filter by means of the impulse invariance method.

Sol. The partial-fraction expansion of $H_a(s)$ is given as

$$H_a(s) = \frac{s+2}{(s+1)(s+3)} = \frac{\frac{1}{2}}{s+1} + \frac{\frac{1}{2}}{s+3}$$

Using eqn. (14) the corresponding digital filter is then

$$H(z) = \frac{1}{2}\left[\frac{1}{1 - e^{-T} z^{-1}} + \frac{1}{1 - e^{-3T} z^{-1}}\right]$$

$$= \frac{1}{2} \frac{2 - z^{-1}(e^{-3T} + e^{-T})}{(1 - e^{-T} z^{-1})(1 - e^{-3T} z^{-1})}$$

$$= \frac{1}{2} \frac{[2 - z^{-1} e^{-2T}(e^{-T} + e^{+T})]}{(1 - e^{-T} z^{-1})(1 - e^{-3T} z^{-1})}$$

$$H(z) = \frac{1 - z^{-1} e^{-2T} \cosh T}{(1 - e^{-T} z^{-1})(1 - e^{-3T} z^{-1})}$$

It should be noted that zero of H(z) at $z = e^{-2T} \cosh T$ is not obtained by transforming the zero at $s = -z$ into a zero at $z = e^{-2T}$.

Problem *Apply the impulse invariant method to obtain the digital filter from the second order analog filter*

$$H_A(s) = \frac{s+a}{(s+a)^2 + b^2}.$$

Sol. The analog filter transfer function is

$$H_A(s) = \frac{s+a}{(s+a+jb)(s+a-jb)}$$

Inverse Laplace transforming,

$$h_A(t) = \begin{cases} e^{-at}\cos(bt), & t \geq 0. \\ 0, & \text{otherwise.} \end{cases}$$

Sampling this function produces

$$h(nT_s) = \begin{cases} e^{-anT_s}\cos(bnT_s), & n \geq 0. \\ 0, & \text{otherwise.} \end{cases}$$

The z-transform of $h(nT_s)$, is equal to

$$H(z) = \sum_{n=0}^{\infty} e^{-anT_s}\cos(b_n T_s)\, z^{-n}$$

$$H(z) = \sum_{n=0}^{\infty} [e^{-aT_s}\cos(bT_s)\, z^{-1}]^n$$

$$H(z) = \frac{1 - e^{-aT}\cos(bT_s)\, z^{-1}}{(1 - e^{-(a+jb)T_s}\, z^{-1})(1 - e^{-(a-jb)T_s}\, z^{-1})}.$$

Video Content / Details of website for further learning (if any):
https://www.alljntuworld.in/download/digital-signal-processing-dsp-materials-notes/#

Important Books/Journals for further learning including the page nos.:
John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 291-294

CourseFaculty

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-17

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.GANESH BABU

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: Approximation of Derivatives

Introduction :

IIR Filter Design by Approximation of Derivatives Analogue filters having rational transfer function H(s) can be described by the linear constant coefficient differential equation. One of the simplest methods for converting an analog filter into a digital filter is to approximate the differential equation by an equivalent difference equation. This approach is often used to solve a linear constant coefficient differential equation numerically on a digital computer

Prerequisite knowledge for Complete understanding and learning of Topic:

1.Infinite Impulse Response

Detailed content of the Lecture:

The 2nd derivative is replaced by the second difference

$$\frac{d^2y(t)}{dt^2} = \frac{d}{dt}\left[\frac{dy(t)}{dt}\right]_{t=nT} = \frac{[y(nT)-y(nt-T)]/T-[y(nT-T)-y(nT-2T)]/T}{T}$$

$$= \frac{y(n)-2y(n-1)+y(n-2)}{T^2}$$

In frequency domain this is equivalent to

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

Then the k^{th} derivative is equivalent frequency-domain relationship

$$s^k = \left(\frac{1-z^{-1}}{T}\right)^k$$

The system function for IIR filter obtained as a result of approximation of the derivatives is

$$H(z) = H_a(s)\Big|_{s=(1-z^{-1})/T}$$

Let us investigate the implication of the mapping from the s-plane to the z-plane

$$z = \frac{1}{1-sT}$$

Then the $j\Omega$ axis is mapped into

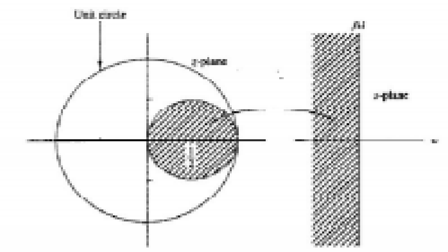
$$z = \frac{1}{1-j\Omega T}$$

$$= \frac{1}{1+\Omega^2 T^2} + j\frac{\Omega T}{1+\Omega^2 T^2}$$

The y-axis in the s-plane is mapped into a circle of radius $\frac{1}{2}$ and with centre z = $\frac{1}{2}$.

Any point in the left hand plane will be mapped to points inside that circle.

So the resultant filter is stable



The possible locations of poles of the digital filter are confined to small frequencies.

As consequence, the mapping is restricted to low-pass and band-pass filters with relatively small resonant frequencies.

It is not possible to transform a high-pass analog filter into a corresponding high-pass digital filter.

In attempt to overcome this limitations, more complex substitutions for the derivatives have been proposed such as

$$\frac{dy(t)}{dt}\Big|_{t=nT} = \frac{1}{T}\sum_{k=1}^{L}\alpha_k \frac{y(nT+kT)-y(nT-kT)}{T}$$

The resulting mapping between the s-plane and the z-plane is now

$$s = \frac{1}{T}\sum_{k=0}^{L}\alpha_k\left(z^k - z^{-k}\right)$$

When $z = e^{j\omega}$

$$s = \frac{j2}{T}\sum_{k=1}^{L}\alpha_k \sin \omega k = j\Omega$$

Which is pure imaginary, which means by carefully selecting α_ks,

The $j\Omega$ axis can be transformed into the unity circle. This can be done by using optimization

Example: Convert the analogue bandpass filter with system function

$$H_a(s) = \frac{1}{(s+0.1)^2+9}$$

Into a digital IIR filter by substituting for the derivatives method

$$H(z) = \frac{1}{((1-z^{-1})/T+0.1)^2+9}$$

$$H(z) = \frac{T^2/(1+0.2T+9.01T^2)}{1-\frac{2(1+0.1T)}{1+0.2T+9.01T^2}z^{-1}+\frac{1}{1+0.2T+9.01T^2}z^{-2}}$$

T should be less than or equal 0.1 for poles near the unity circle

Example: Convert the analogue bandpass filter in the previous example by use of the mapping

$$s = \frac{1}{T}(z - z^{-1})$$

Solution: by substituting for s in $H(s)$, we obtain

$$H(z) = \frac{1}{((z-z^{-1})/T+0.1)^2+9}$$

$$H(z) = \frac{z^2T^2}{z^4+0.2Tz^3+(2+9.01T^2)z^2-0.2Tz+1}$$

$H(z)$ has four poles while $H(s)$ has two poles which means that the conversion has led to more complex system

Video Content / Details of website for further learning (if any):
http://site.iugaza.edu.ps/ahdrouss/files/2010/02/ch10-4.pdf

Important Books/Journals for further learning including the page nos.:
John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 301-309

CourseFaculty

Verified by HOD

LECTURE HANDOUTS

L-18

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.GANESH BABU**

Unit : DESIGN OF IIR FILTER **Date of Lecture :**

Topic of Lecture: Filter design using frequency translation.

Introduction :
Frequency translation is the process of moving a signal from one part of the frequency axis, to another part of the axis. Frequency translation is often done in wireless communications systems to move a pass band signal to base band before demodulation

Prerequisite knowledge for Complete understanding and learning of Topic:

1.Infinite Impulse Response

Detailed content of the Lecture:

Filter design using frequency translation:

FREQUENCY TRANSFORMATION

When the cutoff frequency Ωc of the low pass filter is equal to 1 then it is called normalized filter. Frequency transformation techniques are used to generate High pass filter, Bandpass and bandstop filter from the lowpass filter system function.

FREQUENCY TRANSFORMATION (ANALOG FILTER)

Sr No	Type of transformation	Transformation (Replace s by)
1	Low Pass	$\frac{s}{\omega_{lp}}$ ω_{lp} - Password edge frequency of another LPF
2	High Pass	$\frac{\omega_{hp}}{s}$ ω_{hp} = Password edge frequency of HPF
3	Band Pass	$\frac{(s^2 + \omega_l \omega_h)}{s(\omega_h - \omega_l)}$ ω_h - higher band edge frequency ω_l - Lower band edge frequency
4	Band Stop	$\frac{s(\omega_h - \omega_l)}{s^2 + \omega_h \omega_l}$ ω_h - higher band edge frequency ω_l - Lower band edge frequency

FREQUENCY TRANSFORMATION (DIGITAL FILTER)

Sr No	Type of transformation	Transformation (Replace z^{-1} by)
1	Low Pass	$\frac{z^{-1} - a}{1 - az^{-1}}$
2	High Pass	$\frac{-(z^{-1} + a)}{1 + az^{-1}}$
3	Band Pass	$\frac{-(z^{-2} - a_1 z^{-1} + a_2)}{a_2 z^{-2} - a_1 z^{-1} + 1}$
4	Band Stop	$\frac{z^{-2} - a_1 z^{-1} + a_2}{a_2 z^{-2} - a_1 z^{-1} + 1}$

Example:
Q) Design high pass butterworth filter whose cutoff frequency is 30 Hz at sampling frequency of 150 Hz. Use BZT and Frequency transformation.

Step 1. To find the prewarp cutoff frequency

$$\omega c^{*} = \tan(\omega c Ts/2) = 0.7265$$

Step 2. LPF to HPF transformation

For First order LPF transfer function H(s) = 1/(s+1) Scaled transfer function $H^{*}(s) = H(s)\,|_{s=\omega c^{*}/s}$

$$H^{*}(s) = s/(s + 0.7265)$$

Step 4. Find out the digital filter transfer function. Replace s by (z-1)/(z+1)

$$H(z) = \frac{z-1}{1.7265z - 0.2735}$$

Q) Design second order band pass butterworth filter whose passband of 200 Hz and 300 Hz and sampling frequency is 2000 Hz. Use BZT and Frequency transformation.

Q) Design second order band pass butterworth filter which meet following specification
Lower cutoff frequency = 210 Hz
Upper cutoff frequency = 330 Hz
Sampling Frequency = 960 sps
Use BZT and Frequency transformation.

Video Content / Details of website for further learning (if any):
http://www.kavery.org.in/engg/cse-ecourse/CS2403-DSP.pdf

Important Books/Journals for further learning including the page nos.:
John G. Proakis&DimitrisG.Manolakis" Digital Signal Processing Principles Algorithms & Applications" Pearson Education / Prentice Hall Fourth Edition, 2007 Pg: 310-312

CourseFaculty

Verified by HOD

LECTURE HANDOUTS

L-19

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : DESIGN OF FIR FILTER

Date of Lecture :

Topic of Lecture: Structure of Finite Impulse Response filter

Introduction :

- The term digital filter arises because these filters operate on discrete-time signals
- The term finite impulse response arises because the filter output is computed as a weighted, finite term sum, of past, present, and perhaps future values of the filter input, i.e.,

$$y[n] = \sum_{k=-M_1}^{M_2} b_k x[n-k]$$

where both M1 and M2 are finite

- An FIR filter is based on a feed-forward difference equation
- Feed-forward means that there is no feedback of past or future outputs to form the present output, just input related terms

Prerequisite knowledge for Complete understanding and learning of Topic:

- Filters
- Types of Filters

Detailed content of the Lecture:

- For a FIR design, we have a couple of options such as the windowing method, the frequency sampling method, and more.
- Then, we need to choose a realization structure for the obtained system function. In other words, there are several structures which exhibit the same system function H(z)

- One consideration for choosing the appropriate structure is the sensitivity to coefficient quantization.
- Since a digital filter uses a finite number of bits to represent signals and coefficients, we need structures which can somehow retain the target filter specifications even after quantizing the coefficients.
- In addition, sometimes we observe that a particular structure can dramatically reduce the computational complexity of the system.

- The basic block diagram for an FIR filter of length N is shown below,

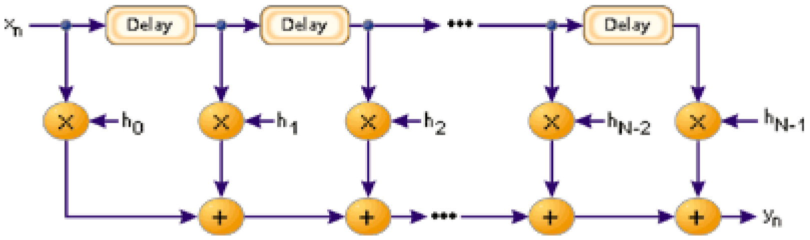


Figure 2. The logical structure of an FIR filter

- The delays result in operating on prior input samples.
- The hk values are the coefficients used for multiplication, so that the output at time n is the summation of all the delayed samples multiplied by the appropriate coefficients.
- The process of selecting the filter's length and coefficients is called filter design.
- The goal is to set those parameters such that certain desired stopband and passband parameters will result from running the filter.

- The results of the design effort should be the same:

  1.A frequency response plot verifies that the filter meets the desired specifications, including ripple and transition bandwidth.

  2.The filter's length and coefficients.

Advantages of FIR filter:

- They can easily be designed to be “linear phase”
- They are simple to implement
- They are suited to multi-rate applications.
- They have desirable numeric properties.

Disadvantages of FIR filter:

Compared to IIR filters, FIR filters sometimes have the disadvantage that they require more memory and/or calculation to achieve a given filter response characteristic. Also, certain responses are not practical to

implement with FIR filters.
Video Content / Details of website for further learning (if any): https://www.youtube.com/watch?v=3OFNS8lxa-0
Important Books/Journals for further learning including the page nos.: https://www.sciencedirect.com/topics/engineering/finite-impulse-response https://www.allaboutcircuits.com/technical-articles/structures-for-implementing-finite-impulse-response-filters/ https://www.vyssotski.ch/BasicsOfInstrumentation/SpikeSorting/Design_of_FIR_Filters.pdf

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L-20

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof/ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Linear phase Finite Impulse Response filter

Introduction :

- Linear phase is a property of a filter, where the phase response of the filter is a linear function of frequency.
- The result is that all frequency components of the input signal are shifted in time (usually delayed) by the same constant amount (the slope of the linear function), which is referred to as the group delay.
- There is no phase distortion due to the time delay of frequencies relative to one another.
- For discrete-time signals, perfect linear phase is easily achieved with a finite impulse response (FIR) filter by having coefficients which are symmetric or anti-symmetric

Prerequisite knowledge for Complete understanding and learning of Topic:

- A filter is called a linear phase filter if the phase component of the frequency response is a linear function of frequency.
- For a continuous-time application, the frequency response of the filter is the Fourier transform of the filter's impulse response, and a linear phase version has the form:

$$H(\omega) = A(\omega)\, e^{-j\omega\tau},$$

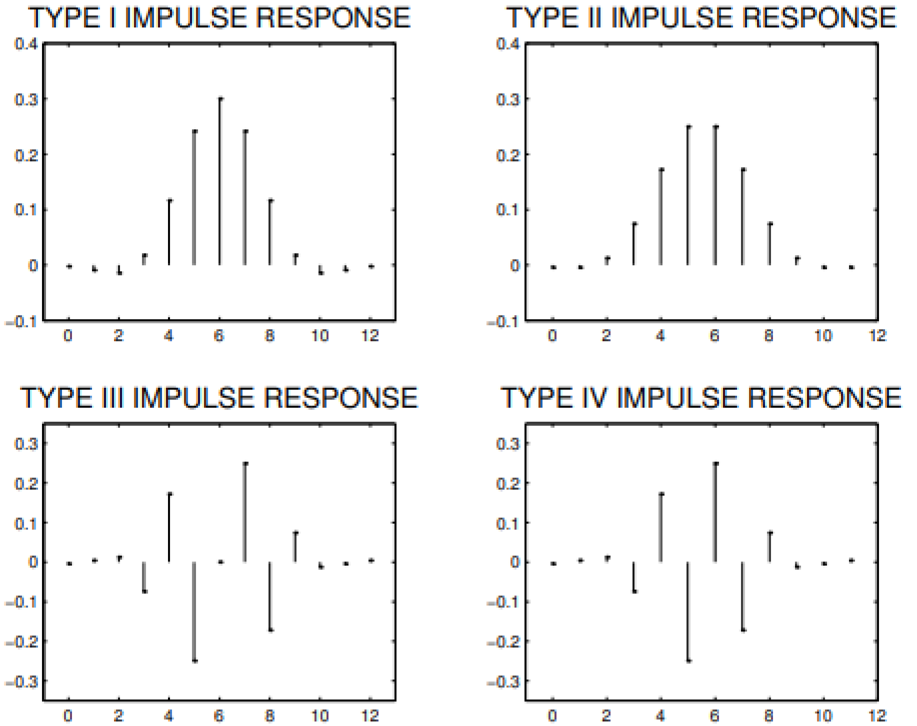
where:

- $A(\omega)$ is a real-valued function.
- τ is the group delay.

Detailed content of the Lecture:

Linear-phase FIR filter can be divided into four basic types.

Type	Impulse response	
I	symmetric	length is odd
II	symmetric	length is even
III	anti-symmetric	length is odd
IV	anti-symmetric	length is even



When h(n) is nonzero for $0 \leq n \leq N-1$ (the length of the impulse response h(n) is N), then the symmetry of the impulse response can be written as $h(n) = h(N-1-n)$ and the anti-symmetry can be written as $h(n) = -h(N-1-n)$.

<u>TYPE I: ODD-LENGTH SYMMETRIC</u>

The frequency response of a length N = 5 FIR Type I filter can be written as follows

$$H^f(\omega) = h_0 + h_1 e^{-j\omega} + h_2 e^{-2j\omega} + h_1 e^{-3j\omega} + h_0 e^{-4\omega} \quad (1)$$

$$= e^{-2j\omega}\left(h_0 e^{2j\omega} + h_1 e^{j\omega} + h_2 + h_1 e^{-j\omega} + h_0 e^{-2j\omega}\right) \quad (2)$$

$$= e^{-2j\omega}\left(h_0(e^{2j\omega} + e^{-2j\omega}) + h_1(e^{j\omega} + e^{-j\omega}) + h_2\right) \quad (3)$$

$$= e^{-2j\omega}\left(2h_0 \cos(2\omega) + 2h_1 \cos(\omega) + h_2\right) \quad (4)$$

$$= A(\omega) e^{j\theta(\omega)} \quad (5)$$

where

$$\theta(\omega) = -2\omega, \qquad A(\omega) = 2h_0 \cos(2\omega) + 2h_1 \cos(\omega) + h_2.$$

Note that $A(\omega)$ is real-valued and can be both positive and negative.

In general, for a Type I FIR filters of length N:

$$H^f(\omega) = A(\omega) e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = h(M) + 2\sum_{n=0}^{M-1} h(n)\cos((M-n)\omega).}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE II: EVEN-LENGTH SYMMETRIC

The frequency response of a length N = 4 FIR Type II filter can be written as follows.

$$
\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} + h_1 e^{-2j\omega} + h_0 e^{-3j\omega} && (6)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} + h_1 e^{-\frac{1}{2}j\omega} + h_0 e^{-\frac{3}{2}j\omega}\right) && (7)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} + e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} + e^{-\frac{1}{2}j\omega})\right) && (8)\\
&= e^{-\frac{3}{2}j\omega}\left(2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right)\right) && (9)\\
&= A(\omega)e^{j\theta(\omega)} && (10)
\end{aligned}
$$

where

$$\theta(\omega) = -\frac{3}{2}\omega, \qquad A(\omega) = 2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type II FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n)\cos\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE III: ODD-LENGTH ANTI-SYMMETRIC

In general, for a Type III FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{M-1} h(n)\sin\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE IV: EVEN-LENGTH ANTI-SYMMETRIC

The frequency response of a length N = 4 FIR Type IV filter can be written as follows.

$$\begin{aligned} H^f(\omega) &= h_0 + h_1 e^{-j\omega} - h_1 e^{-2j\omega} - h_0 e^{-3j\omega} && (18) \\ &= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} - h_1 e^{-\frac{1}{2}j\omega} - h_0 e^{-\frac{3}{2}j\omega}\right) && (19) \\ &= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} - e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} - e^{-\frac{1}{2}j\omega})\right) && (20) \\ &= e^{-\frac{3}{2}j\omega}\left(2jh_0 \sin\left(\tfrac{3}{2}\omega\right) + 2jh_1 \sin\left(\tfrac{1}{2}\omega\right)\right) && (21) \\ &= e^{-\frac{3}{2}j\omega} j \left(2h_0 \sin\left(\tfrac{3}{2}\omega\right) + 2h_1 \sin\left(\tfrac{1}{2}\omega\right)\right) && (22) \\ &= e^{-\frac{3}{2}j\omega} e^{j\frac{\pi}{2}} \left(2h_0 \sin\left(\tfrac{3}{2}\omega\right) + 2h_1 \sin\left(\tfrac{1}{2}\omega\right)\right) && (23) \\ &= A(\omega) e^{j\theta(\omega)} && (24) \end{aligned}$$

where

$$\theta(\omega) = -\frac{3}{2}\omega + \frac{\pi}{2}, \qquad A(\omega) = 2h_0 \sin\left(\tfrac{3}{2}\omega\right) + 2h_1 \sin\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\, e^{j\theta(\omega)}$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\, e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n) \sin\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=KVOkWcknvc4

Important Books/Journals for further learning including the page nos.:

http://eeweb.poly.edu/iselesni/EL713/zoom/linphase.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-21

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Finite Impulse Response filter

Introduction :

- Linear phase is a property of a filter, where the phase response of the filter is a linear function of frequency.
- The result is that all frequency components of the input signal are shifted in time (usually delayed) by the same constant amount (the slope of the linear function), which is referred to as the group delay.
- There is no phase distortion due to the time delay of frequencies relative to one another.
- For discrete-time signals, perfect linear phase is easily achieved with a finite impulse response (FIR) filter by having coefficients which are symmetric or anti-symmetric

Prerequisite knowledge for Complete understanding and learning of Topic:

- A filter is called a linear phase filter if the phase component of the frequency response is a linear function of frequency.
- For a continuous-time application, the frequency response of the filter is the Fourier transform of the filter's impulse response, and a linear phase version has the form:

$$H(\omega) = A(\omega)\, e^{-j\omega\tau},$$

where:

- $A(\omega)$ is a real-valued function.
- τ is the group delay.

Detailed content of the Lecture:

TYPE I: ODD-LENGTH SYMMETRIC

The frequency response of a length N = 5 FIR Type I filter can be written as follows

$$
\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} + h_2 e^{-2j\omega} + h_1 e^{-3j\omega} + h_0 e^{-4\omega} && (1)\\
&= e^{-2j\omega}\left(h_0 e^{2j\omega} + h_1 e^{j\omega} + h_2 + h_1 e^{-j\omega} + h_0 e^{-2j\omega}\right) && (2)\\
&= e^{-2j\omega}\left(h_0(e^{2j\omega} + e^{-2j\omega}) + h_1(e^{j\omega} + e^{-j\omega}) + h_2\right) && (3)\\
&= e^{-2j\omega}\left(2h_0\cos(2\omega) + 2h_1\cos(\omega) + h_2\right) && (4)\\
&= A(\omega)e^{j\theta(\omega)} && (5)
\end{aligned}
$$

where

$$\theta(\omega) = -2\omega, \qquad A(\omega) = 2h_0\cos(2\omega) + 2h_1\cos(\omega) + h_2.$$

Note that $A(\omega)$ is real-valued and can be both positive and negative.

In general, for a Type I FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = h(M) + 2\sum_{n=0}^{M-1} h(n)\cos((M-n)\omega).}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE II: EVEN-LENGTH SYMMETRIC

The frequency response of a length N = 4 FIR Type II filter can be written as follows.

$$
\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} + h_1 e^{-2j\omega} + h_0 e^{-3j\omega} && (6)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} + h_1 e^{-\frac{1}{2}j\omega} + h_0 e^{-\frac{3}{2}j\omega}\right) && (7)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} + e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} + e^{-\frac{1}{2}j\omega})\right) && (8)\\
&= e^{-\frac{3}{2}j\omega}\left(2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right)\right) && (9)\\
&= A(\omega)e^{j\theta(\omega)} && (10)
\end{aligned}
$$

where

$$\theta(\omega) = -\frac{3}{2}\omega, \qquad A(\omega) = 2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type II FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n)\cos\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

<u>TYPE III: ODD-LENGTH ANTI-SYMMETRIC</u>

In general, for a Type III FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{M-1} h(n)\sin((M-n)\omega)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE IV: EVEN-LENGTH ANTI-SYMMETRIC

The frequency response of a length N = 4 FIR Type IV filter can be written as follows.

$$\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} - h_1 e^{-2j\omega} - h_0 e^{-3j\omega} && (18)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} - h_1 e^{-\frac{1}{2}j\omega} - h_0 e^{-\frac{3}{2}j\omega}\right) && (19)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} - e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} - e^{-\frac{1}{2}j\omega})\right) && (20)\\
&= e^{-\frac{3}{2}j\omega}\left(2jh_0\sin\left(\tfrac{3}{2}\omega\right) + 2jh_1\sin\left(\tfrac{1}{2}\omega\right)\right) && (21)\\
&= e^{-\frac{3}{2}j\omega}j\left(2h_0\sin\left(\tfrac{3}{2}\omega\right) + 2h_1\sin\left(\tfrac{1}{2}\omega\right)\right) && (22)\\
&= e^{-\frac{3}{2}j\omega}e^{j\frac{\pi}{2}}\left(2h_0\sin\left(\tfrac{3}{2}\omega\right) + 2h_1\sin\left(\tfrac{1}{2}\omega\right)\right) && (23)\\
&= A(\omega)e^{j\theta(\omega)} && (24)
\end{aligned}$$

where

$$\theta(\omega) = -\frac{3}{2}\omega + \frac{\pi}{2}, \qquad A(\omega) = 2h_0\sin\left(\tfrac{3}{2}\omega\right) + 2h_1\sin\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\,e^{j\theta(\omega)}$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\,e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n)\sin((M-n)\omega)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

Video Content / Details of website for further learning (if any): https://www.youtube.com/watch?v=KVOkWcknvc4
Important Books/Journals for further learning including the page nos.: http://eeweb.poly.edu/iselesni/EL713/zoom/linphase.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-22

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Fourier Series

Introduction :

- Fourier series is, in some way a combination of the Fourier sine and Fourier cosine series.
- The Fourier sine/cosine series are not the series will actually converge to f(x) or not at this point.
- The process of deriving the weights that describe a given function is a form of Fourier analysis. For functions on unbounded intervals, the analysis and synthesis analogies are Fourier transform and inverse transform.

Prerequisite knowledge for Complete understanding and learning of Topic:

A Fourier series is an expansion of a periodic function f(x) in terms of an infinite sum of sines and cosines. Fourier series make use of the orthogonality relationships of the sine and cosine functions. The computation and study of Fourier series is known as harmonic analysis and is extremely useful as a way to break up an arbitrary periodic function into a set of simple terms that can be plugged in, solved individually, and then recombined to obtain the solution to the original problem or an approximation to it to whatever accuracy is desired or practical.The mathematical expression

$$A_0 + \sum_{n=1}^{\infty} (A_n \cos(nx) + B_n \sin(nx)).$$

is called fourier series.

Detailed content of the Lecture:

- One of the successful methods, for the design of FIR filters is based on the application of the Fourier series.
- In this method, it is observed that the frequency of an FIR filter is a periodic function of frequency with a period equal to the sampling frequency and consequently, it can be expressed in terms of the Fourier series.
- The Fourier series by itself does not lead to satisfactory results but by using the Fourier series in conjunction with a special class of functions known as window functions, good results can be obtained.
- Approximation obtained by this method is suboptimal but the amounts of design effort and computation required are relatively insignificant.

The frequency response of an FIR digital filter can be represented by the Fourier series as,

$$H(\omega)=\sum_{n=-\infty}^{\infty} h(n)e^{-j\omega n}$$

where the Fourier coefficients h(n) are the desired impulse response sequence of the filter, which can be determined from

$$h(n)=\frac{1}{2\pi}\int_{0}^{2\pi} H(\omega)\, e^{+j\omega n}\partial\omega .$$

If we substitute e)“ = z, we obtain the transfer function ofthe digital filter, that is,

$$H(z)=\sum_{n=-\infty}^{\infty} h(n)\, z^{-n}.$$

Therefore, if an expression is available for the frequency response, a transfer function can be obtained, which happens to be a non causal and of infinite order. A finite order transfer 35 function can be obtained by truncating the Fourier series. This can be accomplished by letting,

$$h(n)=0 \text{ for} |n| > \frac{(N-1)}{2}$$

$$H(z)=h(0)+\sum_{n=1}^{\frac{(N-1)}{2}}\left[h(-n)z^{n}+h(n)z^{-n}\right].$$

- This modification does not change the amplitude response of the filter, however the abrupt truncation of the Fourier series results in oscillations in the passband and stopband.
- These oscillations are due to slow convergence of the Fourier series, particularly near points of discontinuity. This effect is known as the Gibbs phenomenon

The amplitude of Gibbs oscillations can be reduced using discrete window functions.

A window function, represented by w(n) has the following time-domain properties:

1. w(n) = 0 for | n j> (N-1)/2
2. It is symmetrical, i.e., w(-n) = w(n).

The application of a window function consists of multiplying the impulse response obtained by applying the Fourier series by the window function to obtain a modified impulse response as

$$hw(n) = w(n)\,h(n)$$

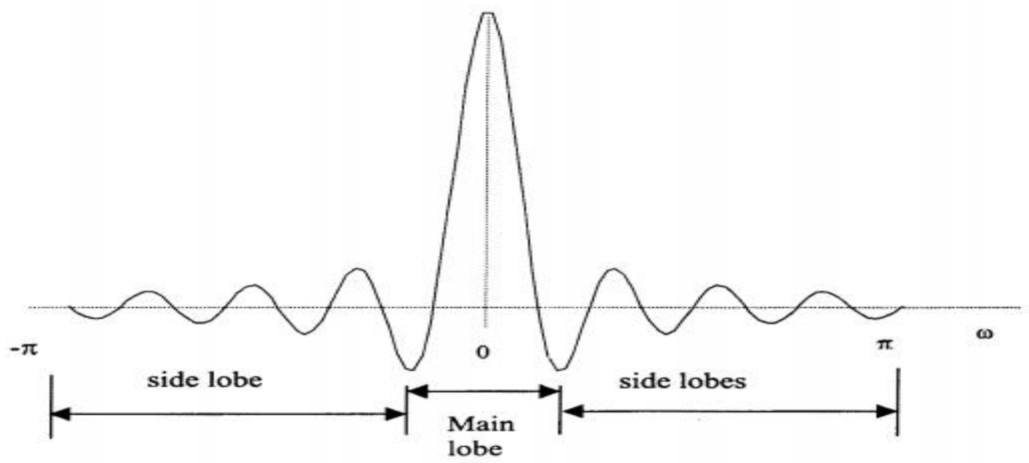


Fig. 2.2 Spectrum of typical window function

Since the window function is of finite duration, a finite-order transfer function given by

$$H_w(z) = \sum_{n=-\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} w(n)\,h(n)\,z^{-n}$$

Through the application of the complex convolution, the frequency response of the modified filter can be expressed as,

$$H_w(\omega) = \frac{1}{2\pi} \int_0^{2\pi} H(\Omega)\ w(e^{j(\omega-\Omega)})\, \partial\Omega$$

The application of the window function has two effects on the amplitude response of the filter. First, the amplitudes of Gibbs' oscillations in the pass bands and stop bands are directly related to the ripple ratio of the window. Second, transition bands are introduced between pass bands and stop bands whose width is directly related to the main-lobe width of the window.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=nDdX11EkOJ0

Important Books/Journals for further learning including the page nos.:

https://shodhganga.inflibnet.ac.in/bitstream/10603/95357/9/09_chapter%202.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-23

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique

Introduction :

- There are essentially three well-known methods for FIR filter design namely:

  (1) The window method

  (2) The frequency sampling technique

  (3) Optimal filter design methods

<u>The Window Method:</u>

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

- The window method for digital filter design is fast, convenient, and robust, but generally suboptimal.
- It is easily understood in terms of the convolution theorem for Fourier transforms, making it instructive to study after the Fourier theorems and windows for spectrum analysis.
- It can be effectively combined with the frequency sampling method.
- The window method consists of simply ``windowing'' a theoretically ideal filter impulse

response $h(n)$ by some suitably chosen window function $w(n)$,

$$h_w(n) = w(n) \cdot h(n), \quad n \in \mathcal{Z}.$$

- Window functions are always *time* limited. This means there is always a finite integer N_w such that $w(n) = 0$ for all $|n| > N_w$.

  The final windowed impulse response $h_w(n) = w(n) \cdot h(n)$ is thus always time-limited

Detailed content of the Lecture:

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

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

The unit sample response hd(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 hd(n) to length M-1 is same as multiplying hd(n) by the rectangular window defined as

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

Thus the unit sample response of the FIR filter becomes

$$\begin{aligned} h(n) &= h_d(n)\, w(n) \\ &= h_d(n) \qquad 0 \le n \le M\text{-}1 \\ &= 0 \qquad \text{otherwise} \end{aligned}$$

The multiplication of the window function w(n) with hd(n) is equivalent to convolution of Hd(w) with W(w), where W(w) is the frequency domain representation of the window function

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

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

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

The frequency response can also be obtained using the following relation

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

But direct truncation of hd(n) to M terms to obtain h(n) leads to the Gibbs phenomenon effect which manifests itself as a fixed percentage overshoot and ripple before and after an approximated discontinuity in the frequency response due to the non-uniform convergence of the fourier series at a discontinuity. Thus the frequency response obtained by using (8) contains 3 ripples in the frequency domain. In order to reduce the ripples, instead of multiplying hd(n) with a rectangular window w(n), hd(n) is multiplied with a window function.

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

(i) A major effect is that discontinuities in H(w) become transition bands between values on either side of the discontinuity.

(ii) 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).

(iii) Since the filter frequency response is obtained via a convolution relation, it is clear that the resulting filters are never optimal in any sense.

(iv) As M (the length of the window function) increases, the main lobe width of W(w) is reduced which reduces the width of the transition band, but this also introduces more ripple in the frequency response.

(v) The window function eliminates the ringing effects at the band edge and does result in lower side lobes at the expense of an increase in the width of the transition band of the filter.

Video Content / Details of website for further learning (if any):
https://www.youtube.com/watch?v=KVOkWcknvc4

Important Books/Journals for further learning including the page nos.:
https://www.ee.iitb.ac.in/~esgroup/es_mtech02_sem/es02_sem_rep_arojit.pdf
https://www.allaboutcircuits.com/technical-articles/finite-impulse-response-filter-design-by-windowing-part-i-concepts-and-rect/

Course Teacher

Verified by HOD

LECTURE HANDOUTS

L-24

ECE

III/V

Course Name with Code : 16ECD09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : FIR FILTER DESIGN **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique-Rectangular window

Introduction :

- There are essentially three well-known methods for FIR filter design namely:
  (1) The window method
  (2) The frequency sampling technique
  (3) Optimal filter design methods

The Window Method:

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

- A window is a finite array consisting of coefficients selected to satisfy the desirable requirements.
- While designing digital FIR filter using window function it is necessary to specify a window function to be used and the filter order according to the required specifications (selectivity and stop band attenuation).
- These two requirements are interrelated .

- Window Technique implicates a function called window Function.
- It is also known as Tapering Function.

Detailed content of the Lecture:

Rectangular Window

The rectangular window (sometimes known as the Boxer or Dirichlet window) is the simplest window, equivalent to replacing all but N values of a data sequence by zeros, making it appear as though the waveform suddenly turns on and off.

$$w_R(n) \triangleq \begin{cases} 1, & -\frac{M-1}{2} \leq n \leq \frac{M-1}{2} \\ 0, & \text{otherwise} \end{cases}$$

where M is the window length in samples (assumed odd for now). A plot of the rectangular window appears in Fig.3.1 for length $M = 21$. It is sometimes convenient to define windows so that their dc gain is 1, in which case we would multiply the definition above by $1/M$

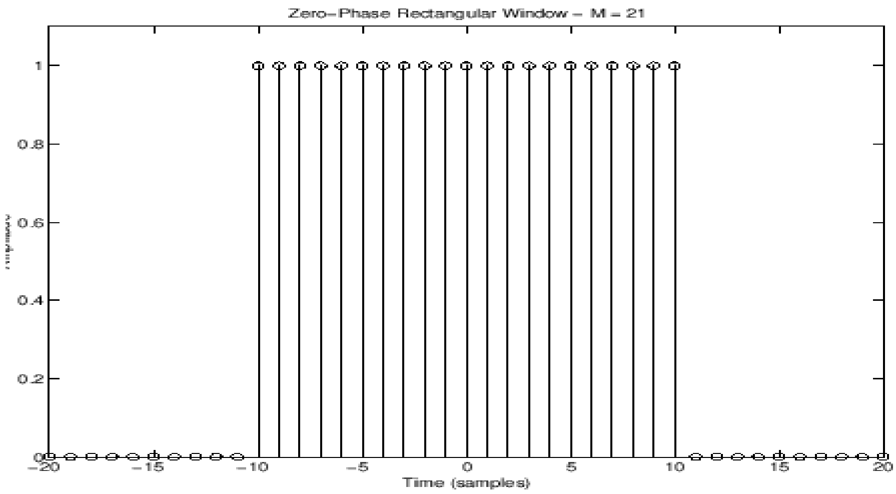


Figure 3.1: The rectangular window.

$$\begin{aligned} W_R(\omega) &= \text{DTFT}_\omega(w_R) \triangleq \sum_{n=-\infty}^{\infty} w_R(n)e^{-j\omega n}, \quad \omega \in [-\pi, \pi) \\ &= \sum_{n=-\frac{M-1}{2}}^{\frac{M-1}{2}} e^{-j\omega n} = \frac{e^{j\omega\frac{M-1}{2}} - e^{-j\omega\frac{M+1}{2}}}{1 - e^{-j\omega}} \end{aligned}$$

$$W_R(\omega) = \frac{e^{-j\omega\frac{1}{2}}}{e^{-j\omega\frac{1}{2}}}\left[\frac{e^{j\omega\frac{M}{2}} - e^{-j\omega\frac{M}{2}}}{e^{j\omega\frac{1}{2}} - e^{-j\omega\frac{1}{2}}}\right]$$

$$= \frac{\sin\left(M\frac{\omega}{2}\right)}{\sin\left(\frac{\omega}{2}\right)} \triangleq M \cdot \text{asinc}_M(\omega)$$

$$\text{asinc}_M(\omega) \triangleq \frac{\sin(M\omega/2)}{M \cdot \sin(\omega/2)}$$

also called the Dirichlet function or periodic sinc function. This (real) result is for the zero-centered rectangular window. For the causal case, a linear phase term appears:

$$W_R^c(\omega) = e^{-j\frac{M-1}{2}\omega} \cdot M \cdot \text{asinc}_M(\omega)$$

The term ``aliased sinc function" refers to the fact that it may be simply obtained by sampling the length- τ continuous-time rectangular window, which has Fourier transform sinc $(f\tau) \triangleq \sin(\pi f\tau)/(\pi f\tau)$ (given amplitude $1/\tau$ in the time domain). Sampling at intervals of T seconds in the time domain corresponds to aliasing in the frequency domain over the interval $[0, 1/T]$ Hz, and by direct derivation. The window duration increases continuously in the time domain: the magnitude spectrum can only change in discrete jumps as new samples are included, even though it is continuously parametrized in τ .As the sampling rate goes to infinity, the aliased sinc function therefore approaches the sinc function.

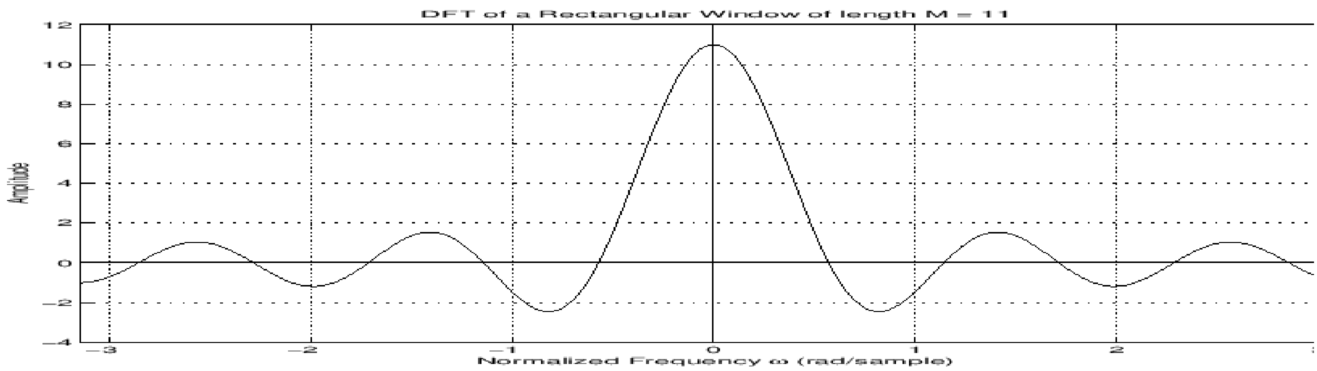


Figure: Fourier transform of the rectangular window.

The phase of rectangular-window transform $W_R(\omega)$ is zero for $|\omega| < 2\pi/M$, which is the width of the main lobe. This is why zero-centered windows are often called zero-phase windows; while the phase actually alternates between 0 and π radians, the π values occur only within side-lobes which are routinely neglected.

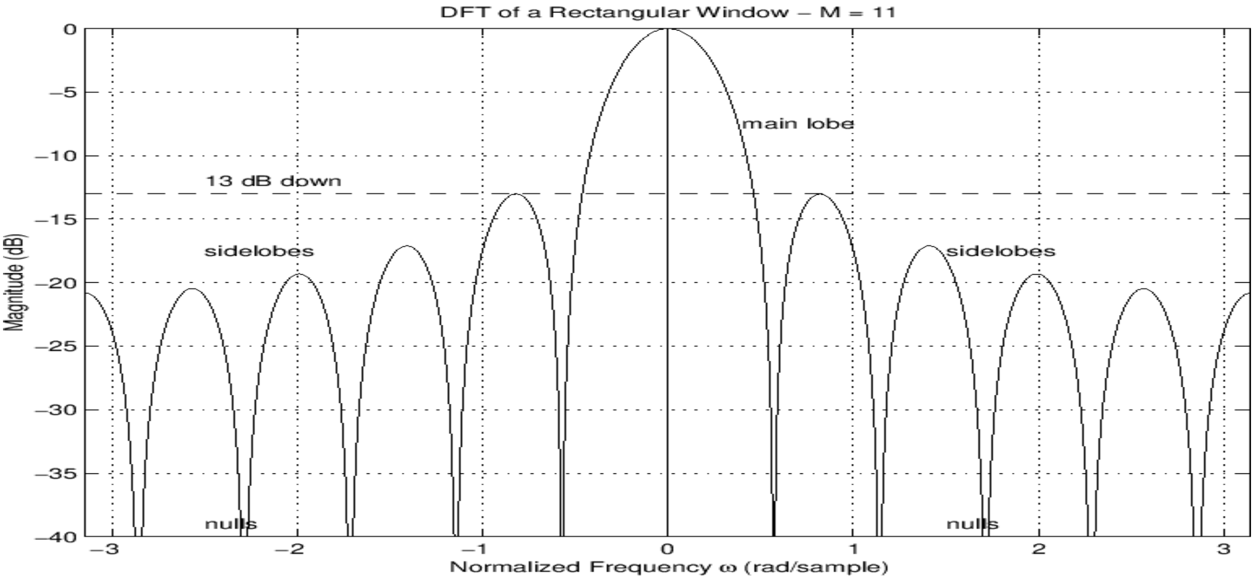


Figure: Magnitude (dB) of the rectangular-window transform

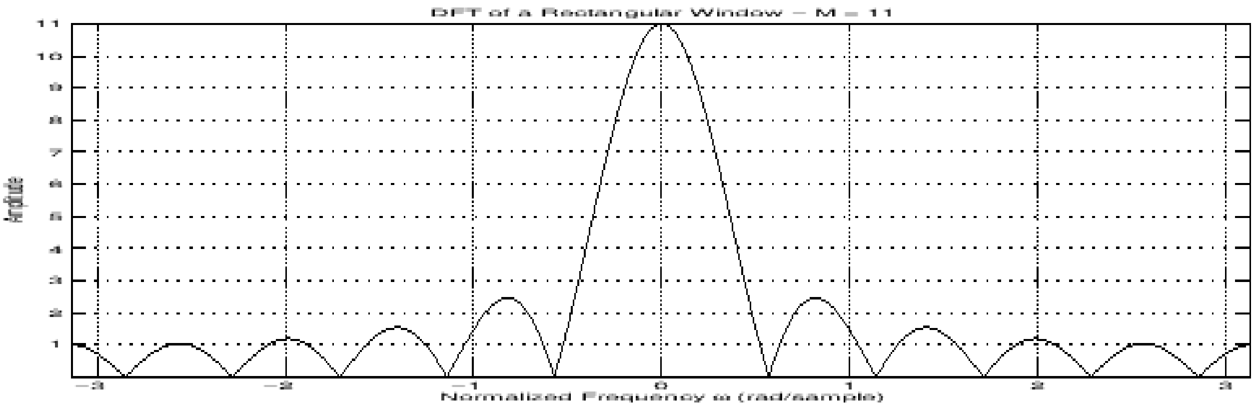


Figure: Magnitude of the rectangular-window Fourier transform.

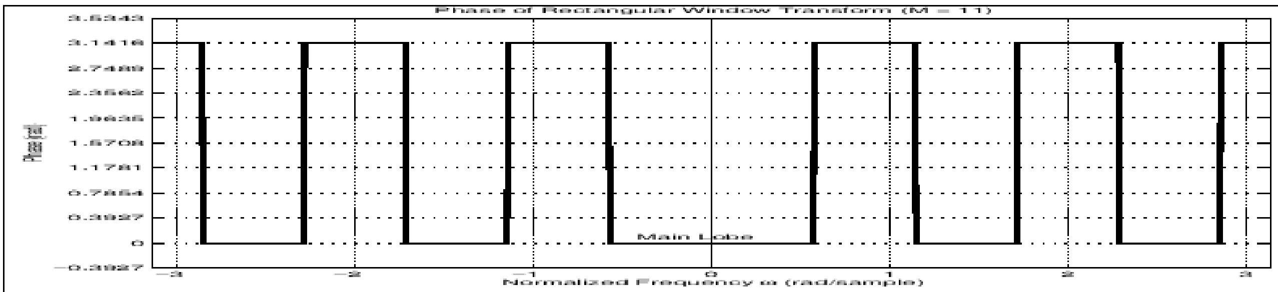


Figure: Phase of the rectangular-window Fourier transform.

Video Content / Details of website for further learning (if any):
https://www.youtube.com/watch?v=s1-RMohyJ1A

Important Books/Journals for further learning including the page nos.:
https://ccrma.stanford.edu/~jos/sasp/Rectangular_Window.html
https://www.dsprelated.com/freebooks/sasp/Rectangular_Window.html

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

IQAC

LECTURE HANDOUTS

L-25

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique-Hamming window

Introduction :

- There are essentially three well-known methods for FIR filter design namely:

  (1) The window method

  (2) The frequency sampling technique

  (3) Optimal filter design methods

<u>The Window Method:</u>

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

- A window is a finite array consisting of coefficients selected to satisfy the desirable requirements.
- While designing digital FIR filter using window function it is necessary to specify a window function to be used and the filter order according to the required specifications (selectivity and stop band attenuation).
- These two requirements are interrelated .
- Window Technique implicates a function called window Function.

- It is also known as Tapering Function.

Detailed content of the Lecture:

Hamming Window

The Hamming window is determined by choosing α (with $\beta \triangleq (1-\alpha)/2$) to cancel the largest side lobe. Doing this results in the values

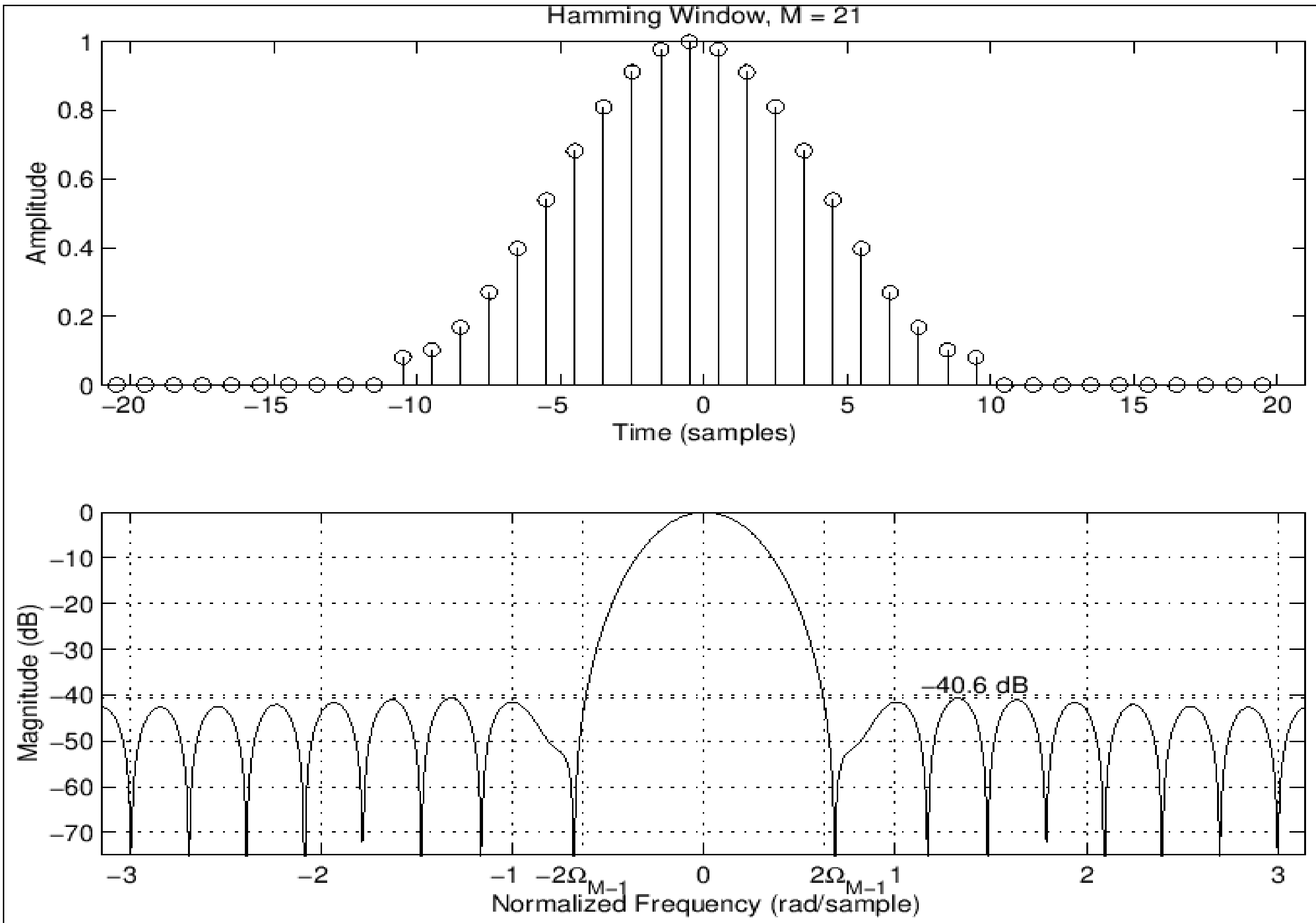
$$\begin{aligned} \alpha &= \frac{25}{46} \approx 0.54 \\ \beta &= \frac{1-\alpha}{2} \approx 0.23. \end{aligned}$$

The peak side-lobe level is approximately -42.76 dB for the Hamming window . It happens that this choice is very close to that which minimizes peak side-lobe level (down to -43.19 dB--the lowest possible within the generalized Hamming family).

$$\alpha = 0.53836\ldots$$

Since rounding the optimal α to two significant digits gives 0.54 , the Hamming window can be considered the ``Chebyshev Generalized Hamming Window" (approximately). Chebyshev-type designs normally exhibit equiripple error behavior, because the worst-case error (side-lobe level in this case) is minimized. Generalized Hamming windows can have a step discontinuity at their endpoints, but no impulsive points.

- The Hamming window and its DTFT magnitude are shown in Figure. The Hamming window is also one period of a raised cosine.
- However, the cosine is raised so high that its negative peaks are above zero, and the window has a discontinuity in amplitude at its endpoints (stepping discontinuously from 0.08 to 0).
- This is 10 dB better than the Hann case of Figure and 28 dB better than the rectangular window.
- The main lobe is approximately wide, as is the case for all members of the generalized Hamming family



For the Hamming window, the side-lobes nearest the main lobe have been strongly shaped by the optimization. As a result, the nearly -6 dB per octave roll-off occurs only over an interior interval of the spectrum, well between the main lobe and half the sampling rate. The optimized side-lobes nearest the main lobe occupy a smaller frequency interval about the main lobe.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=4e_ffrF6HT4

Important Books/Journals for further learning including the page nos.:

https://ccrma.stanford.edu/~jos/sasp/Hamming_Window.html

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-26

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique-Hanning window

Introduction :

- There are essentially three well-known methods for FIR filter design namely:

  (1) The window method

  (2) The frequency sampling technique

  (3) Optimal filter design methods

The Window Method:

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

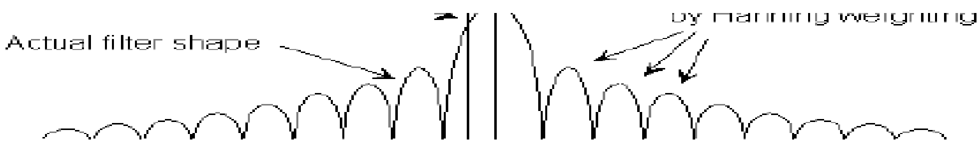
- A window is a finite array consisting of coefficients selected to satisfy the desirable requirements.
- While designing digital FIR filter using window function it is necessary to specify a window function to be used and the filter order according to the required specifications (selectivity and stop band attenuation).

- These two requirements are interrelated .
- Window Technique implicates a function called window Function.
- It is also known as Tapering Function.

Detailed content of the Lecture:

Hanning Window

- Hanning window is the shape of one cycle of a cosine wave with 1 added to it so it is always positive.
- The sampled signal values are multiplied by the Hanning function.
- The ends of the time record are forced to zero regardless of what the input signal is doing.
- While the Hanning window does a good job of forcing the ends to zero, it also adds distortion to the wave form being analyzed in the form of amplitude modulation; i.e., the variation in amplitude of the signal over the time record.
- Amplitude Modulation in a wave form results in sidebands in its spectrum, and in the case of the Hanning window, these sidebands, or side lobes as they are called, effectively reduce the frequency resolution of the analyzer by 50%.
- It is as if the analyzer frequency "lines" are made wider.
- The highest-level side lobes are about 32 dB down from the main lobe.



- The Hanning window should always be used with continuous signals, but must never be used with transients. The reason is that the window shape will distort the shape of the transient, and the frequency and phase content of a transient is intimately connected with its shape.
- The measured level will also be greatly distorted. Even if the transient were in the center of the Hanning window, the measured level would be twice as great as the actual level because of the amplitude correction the analyzer applies when using the Hanning weighting.
- A Hanning weighted signal actually is only half there, the other half of it having been removed by the windowing. This is not a problem with a perfectly smooth and continuous signal like a sinusoid, but most signals we want to analyze, such as machine vibration signatures are not perfectly smooth.
- If a small change occurs in the signal near the beginning or end of the time record, it will either be analyzed at a much lower level than its true level, or it may be missed altogether.

- For this reason, it is a good idea to employ overlap processing. To do this, two time buffers are required in the analyzer.
- For 50% overlap, the sequence of events is as follows:
- When the first buffer is half full, i.e., it contains half the samples of a time record, the second buffer is connected to the data stream and also begins to collect samples. As soon as the first buffer is full, the FFT is calculated, and the buffer begins to take data again.
- When the second buffer is filled, the FFT is again calculated on its contents, and the result sent to the spectrum-averaging buffer. This process continues on until the desired number of averages is collected.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=s3sOhj7DaFg

Important Books/Journals for further learning including the page nos.:

http://www.azimadli.com/vibman/thehanningwindow.htm

Course Teacher

Verified by HOD

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Frequency sampling method

Introduction :

- The frequency-sampling method for FIR filter design is perhaps the simplest and most direct technique imaginable when a desired frequency response has been specified.
- It consists simply of uniformly sampling the desired frequency response, and performing an inverse DFT to obtain the corresponding (finite) impulse response.
- The results are not optimal, however, because the response generally deviates from what is desired between the samples.
- When the desired frequency-response is undersampled, which is typical, the resulting impulse response will be time aliased to some extent.

Prerequisite knowledge for Complete understanding and learning of Topic:

- The main idea of the frequency sampling design method is that a desired frequency response can be approximated by sampling it of N evenly spaced points and then obtaining an interpolated frequency response that passes through the frequency samples.
- For filters with reasonably smooth frequency responses,the interpolation error is generally small.
- In the case of band select filters, where the desired frequency response changes radically across bands, the frequency samples which occur in transition bands are made to be unspecified variables whose values are chosen by an optimization algorithm which minimizes some function of the

approximation error of the filter finally, it was shown that there were two distinct types of frequency sampling filters, depending on where the initial frequency sample.

Detailed content of the Lecture:

The frequency sampling method allows us to design recursive and nonrecursive FIR filters for both standard frequency selective and filters with arbitrary frequency response.

No recursive frequency sampling filters :

The problem of FIR filter design is to find a finite–length impulse response h (n) that corresponds to desired frequency response. In this method h (n) can be determined by uniformly sampling, the desired frequency response HD (ω) at the N points and finding its inverse DFT of the frequency samples

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

where H (k), k = 0, 1, 2,……., N-1, are samples of the HD (ω) . For linear phase filters, with positive symmetrical impulse response, we can write

$$h(n) = \frac{1}{N}\left[\sum_{k=1}^{(N/2)-1} 2|H(k)|\cos\left[\frac{2\pi k(n-\alpha)}{N}\right] + H(0)\right]$$

where α = (N-1)/2. For N odd, the upper limit in the summation is (N - 1)/2, to obtain a good approximation to the desired frequency

Recursive frequency sampling filter :

In recursive frequency sampling method the DFT samples H (k) for an FIR sequence can be regarded as samples of the filters z– transform, evaluated at N points equally spaced around the unit circle.

$$H(k) = H(z)\big|_{z=e^{j(2\pi/N)k}}$$

thus the z–transform of an FIR filter can easily be expressed in terms of its DFT coefficients,

$$H(z) = \sum_{n=0}^{N-1} h(n) z^{-n} = \sum_{n=0}^{N-1}\left[\frac{1}{N}\sum_{k=0}^{N-1} H(k) e^{j(2\pi/N)nk}\right] z^{-n}$$

$$= \sum_{k=0}^{N-1} \frac{H(k)}{N} \sum_{n=0}^{N-1} \left[e^{j(2\pi/N)k} z^{-1} \right]^n$$

$$= \sum_{k=0}^{N-1} \frac{H(k)}{N} \frac{(1 - e^{j2\pi k} z^{N})}{(1 - e^{j(2\pi/N)k} z^{-1})}$$

by butting $e^{j2\pi k} = 1$, Equation (13) reduces to

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

This is the desired result

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=O0DbpZqhPSg

Important Books/Journals for further learning including the page nos.:

https://bulletin.zu.edu.ly/issue_n15_3/Contents/E_04.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-19

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : DESIGN OF FIR FILTER **Date of Lecture :**

Topic of Lecture: Structure of Finite Impulse Response filter

Introduction :

- The term digital filter arises because these filters operate on discrete-time signals
- The term finite impulse response arises because the filter output is computed as a weighted, finite term sum, of past, present, and perhaps future values of the filter input, i.e.,

$$y[n] = \sum_{k=-M_1}^{M_2} b_k x[n-k]$$

where both M1 and M2 are finite

- An FIR filter is based on a feed-forward difference equation
- Feed-forward means that there is no feedback of past or future outputs to form the present output, just input related terms

Prerequisite knowledge for Complete understanding and learning of Topic:

- Filters
- Types of Filters

Detailed content of the Lecture:

- For a FIR design, we have a couple of options such as the windowing method, the frequency sampling method, and more.
- Then, we need to choose a realization structure for the obtained system function. In other words, there are several structures which exhibit the same system function H(z)

- One consideration for choosing the appropriate structure is the sensitivity to coefficient quantization.
- Since a digital filter uses a finite number of bits to represent signals and coefficients, we need structures which can somehow retain the target filter specifications even after quantizing the coefficients.
- In addition, sometimes we observe that a particular structure can dramatically reduce the computational complexity of the system.

- The basic block diagram for an FIR filter of length N is shown below,

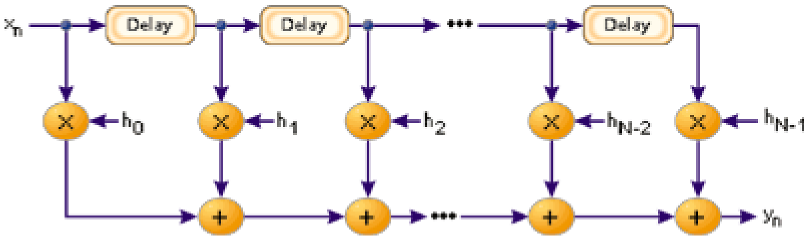


Figure 2. The logical structure of an FIR filter

- The delays result in operating on prior input samples.
- The hk values are the coefficients used for multiplication, so that the output at time n is the summation of all the delayed samples multiplied by the appropriate coefficients.
- The process of selecting the filter's length and coefficients is called filter design.
- The goal is to set those parameters such that certain desired stopband and passband parameters will result from running the filter.

- The results of the design effort should be the same:

  1.A frequency response plot verifies that the filter meets the desired specifications, including ripple and transition bandwidth.

  2.The filter's length and coefficients.

Advantages of FIR filter:

- They can easily be designed to be “linear phase”
- They are simple to implement
- They are suited to multi-rate applications.
- They have desirable numeric properties.

Disadvantages of FIR filter:

Compared to IIR filters, FIR filters sometimes have the disadvantage that they require more memory and/or calculation to achieve a given filter response characteristic. Also, certain responses are not practical to

implement with FIR filters.
Video Content / Details of website for further learning (if any): https://www.youtube.com/watch?v=3OFNS8lxa-0
Important Books/Journals for further learning including the page nos.: https://www.sciencedirect.com/topics/engineering/finite-impulse-response https://www.allaboutcircuits.com/technical-articles/structures-for-implementing-finite-impulse-response-filters/ https://www.vyssotski.ch/BasicsOfInstrumentation/SpikeSorting/Design_of_FIR_Filters.pdf

Course Teacher

Verified by HOD

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof/ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Linear phase Finite Impulse Response filter

Introduction :

- Linear phase is a property of a filter, where the phase response of the filter is a linear function of frequency.
- The result is that all frequency components of the input signal are shifted in time (usually delayed) by the same constant amount (the slope of the linear function), which is referred to as the group delay.
- There is no phase distortion due to the time delay of frequencies relative to one another.
- For discrete-time signals, perfect linear phase is easily achieved with a finite impulse response (FIR) filter by having coefficients which are symmetric or anti-symmetric

Prerequisite knowledge for Complete understanding and learning of Topic:

- A filter is called a linear phase filter if the phase component of the frequency response is a linear function of frequency.
- For a continuous-time application, the frequency response of the filter is the Fourier transform of the filter's impulse response, and a linear phase version has the form:

$$H(\omega) = A(\omega)\, e^{-j\omega\tau},$$

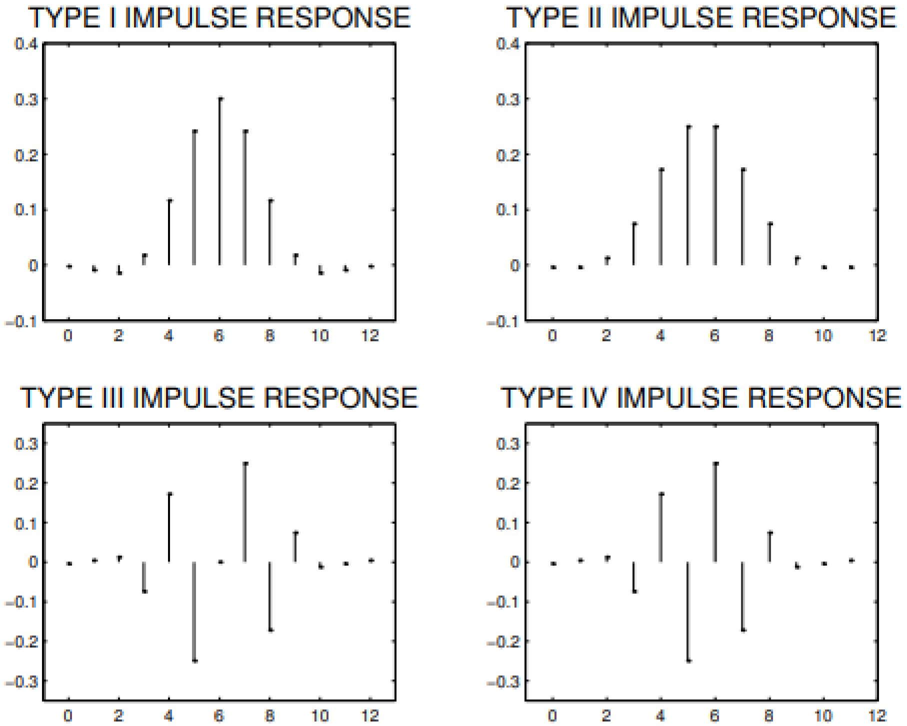
where:

- $A(\omega)$ is a real-valued function.
- τ is the group delay.

Detailed content of the Lecture:

Linear-phase FIR filter can be divided into four basic types.

Type	Impulse response	
I	symmetric	length is odd
II	symmetric	length is even
III	anti-symmetric	length is odd
IV	anti-symmetric	length is even



When h(n) is nonzero for $0 \leq n \leq N-1$ (the length of the impulse response h(n) is N), then the symmetry of the impulse response can be written as $h(n) = h(N-1-n)$ and the anti-symmetry can be written as $h(n) = -h(N-1-n)$.

<u>TYPE I: ODD-LENGTH SYMMETRIC</u>

The frequency response of a length N = 5 FIR Type I filter can be written as follows

$$
\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} + h_2 e^{-2j\omega} + h_1 e^{-3j\omega} + h_0 e^{-4\omega} && (1)\\
&= e^{-2j\omega}\left(h_0 e^{2j\omega} + h_1 e^{j\omega} + h_2 + h_1 e^{-j\omega} + h_0 e^{-2j\omega}\right) && (2)\\
&= e^{-2j\omega}\left(h_0(e^{2j\omega} + e^{-2j\omega}) + h_1(e^{j\omega} + e^{-j\omega}) + h_2\right) && (3)\\
&= e^{-2j\omega}\left(2h_0 \cos(2\omega) + 2h_1 \cos(\omega) + h_2\right) && (4)\\
&= A(\omega) e^{j\theta(\omega)} && (5)
\end{aligned}
$$

where

$$\theta(\omega) = -2\omega, \qquad A(\omega) = 2h_0 \cos(2\omega) + 2h_1 \cos(\omega) + h_2.$$

Note that $A(\omega)$ is real-valued and can be both positive and negative.

In general, for a Type I FIR filters of length N:

$$H^f(\omega) = A(\omega) e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = h(M) + 2\sum_{n=0}^{M-1} h(n)\cos((M-n)\omega).}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

<u>TYPE II: EVEN-LENGTH SYMMETRIC</u>

The frequency response of a length N = 4 FIR Type II filter can be written as follows.

$$
\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} + h_1 e^{-2j\omega} + h_0 e^{-3j\omega} && (6)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} + h_1 e^{-\frac{1}{2}j\omega} + h_0 e^{-\frac{3}{2}j\omega}\right) && (7)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} + e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} + e^{-\frac{1}{2}j\omega})\right) && (8)\\
&= e^{-\frac{3}{2}j\omega}\left(2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right)\right) && (9)\\
&= A(\omega)e^{j\theta(\omega)} && (10)
\end{aligned}
$$

where

$$\theta(\omega) = -\frac{3}{2}\omega, \qquad A(\omega) = 2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type II FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n)\cos\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE III: ODD-LENGTH ANTI-SYMMETRIC

In general, for a Type III FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{M-1} h(n)\sin\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE IV: EVEN-LENGTH ANTI-SYMMETRIC

The frequency response of a length N = 4 FIR Type IV filter can be written as follows.

$$
\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} - h_1 e^{-2j\omega} - h_0 e^{-3j\omega} && (18)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} - h_1 e^{-\frac{1}{2}j\omega} - h_0 e^{-\frac{3}{2}j\omega}\right) && (19)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} - e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} - e^{-\frac{1}{2}j\omega})\right) && (20)\\
&= e^{-\frac{3}{2}j\omega}\left(2jh_0 \sin\left(\tfrac{3}{2}\omega\right) + 2jh_1 \sin\left(\tfrac{1}{2}\omega\right)\right) && (21)\\
&= e^{-\frac{3}{2}j\omega} j \left(2h_0 \sin\left(\tfrac{3}{2}\omega\right) + 2h_1 \sin\left(\tfrac{1}{2}\omega\right)\right) && (22)\\
&= e^{-\frac{3}{2}j\omega} e^{j\frac{\pi}{2}} \left(2h_0 \sin\left(\tfrac{3}{2}\omega\right) + 2h_1 \sin\left(\tfrac{1}{2}\omega\right)\right) && (23)\\
&= A(\omega) e^{j\theta(\omega)} && (24)
\end{aligned}
$$

where

$$\theta(\omega) = -\frac{3}{2}\omega + \frac{\pi}{2}, \qquad A(\omega) = 2h_0 \sin\left(\tfrac{3}{2}\omega\right) + 2h_1 \sin\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\, e^{j\theta(\omega)}$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\, e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n) \sin\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=KVOkWcknvc4

Important Books/Journals for further learning including the page nos.:

http://eeweb.poly.edu/iselesni/EL713/zoom/linphase.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-21

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : DESIGN OF FIR FILTER **Date of Lecture :**

Topic of Lecture: Finite Impulse Response filter

Introduction :

- Linear phase is a property of a filter, where the phase response of the filter is a linear function of frequency.
- The result is that all frequency components of the input signal are shifted in time (usually delayed) by the same constant amount (the slope of the linear function), which is referred to as the group delay.
- There is no phase distortion due to the time delay of frequencies relative to one another.
- For discrete-time signals, perfect linear phase is easily achieved with a finite impulse response (FIR) filter by having coefficients which are symmetric or anti-symmetric

Prerequisite knowledge for Complete understanding and learning of Topic:

- A filter is called a linear phase filter if the phase component of the frequency response is a linear function of frequency.
- For a continuous-time application, the frequency response of the filter is the Fourier transform of the filter's impulse response, and a linear phase version has the form:

$$H(\omega) = A(\omega)\, e^{-j\omega\tau},$$

where:

- $A(\omega)$ is a real-valued function.
- τ is the group delay.

Detailed content of the Lecture:

TYPE I: ODD-LENGTH SYMMETRIC

The frequency response of a length N = 5 FIR Type I filter can be written as follows

$$\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} + h_2 e^{-2j\omega} + h_1 e^{-3j\omega} + h_0 e^{-4\omega} && (1)\\
&= e^{-2j\omega}\left(h_0 e^{2j\omega} + h_1 e^{j\omega} + h_2 + h_1 e^{-j\omega} + h_0 e^{-2j\omega}\right) && (2)\\
&= e^{-2j\omega}\left(h_0(e^{2j\omega} + e^{-2j\omega}) + h_1(e^{j\omega} + e^{-j\omega}) + h_2\right) && (3)\\
&= e^{-2j\omega}\left(2h_0 \cos(2\omega) + 2h_1 \cos(\omega) + h_2\right) && (4)\\
&= A(\omega)e^{j\theta(\omega)} && (5)
\end{aligned}$$

where

$$\theta(\omega) = -2\omega, \qquad A(\omega) = 2h_0 \cos(2\omega) + 2h_1 \cos(\omega) + h_2.$$

Note that $A(\omega)$ is real-valued and can be both positive and negative.

In general, for a Type I FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = h(M) + 2\sum_{n=0}^{M-1} h(n)\cos((M-n)\omega).}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE II: EVEN-LENGTH SYMMETRIC

The frequency response of a length N = 4 FIR Type II filter can be written as follows.

$$H^f(\omega) = h_0 + h_1 e^{-j\omega} + h_1 e^{-2j\omega} + h_0 e^{-3j\omega} \quad (6)$$
$$= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} + h_1 e^{-\frac{1}{2}j\omega} + h_0 e^{-\frac{3}{2}j\omega}\right) \quad (7)$$
$$= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} + e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} + e^{-\frac{1}{2}j\omega})\right) \quad (8)$$
$$= e^{-\frac{3}{2}j\omega}\left(2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right)\right) \quad (9)$$
$$= A(\omega)e^{j\theta(\omega)} \quad (10)$$

where

$$\theta(\omega) = -\frac{3}{2}\omega, \qquad A(\omega) = 2h_0\cos\left(\tfrac{3}{2}\omega\right) + 2h_1\cos\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type II FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n)\cos((M-n)\omega)}$$

$$\boxed{\theta(\omega) = -M\omega}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE III: ODD-LENGTH ANTI-SYMMETRIC

In general, for a Type III FIR filters of length N:

$$H^f(\omega) = A(\omega)e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{M-1} h(n)\sin\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

TYPE IV: EVEN-LENGTH ANTI-SYMMETRIC

The frequency response of a length N = 4 FIR Type IV filter can be written as follows.

$$\begin{aligned}
H^f(\omega) &= h_0 + h_1 e^{-j\omega} - h_1 e^{-2j\omega} - h_0 e^{-3j\omega} && (18)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0 e^{\frac{3}{2}j\omega} + h_1 e^{\frac{1}{2}j\omega} - h_1 e^{-\frac{1}{2}j\omega} - h_0 e^{-\frac{3}{2}j\omega}\right) && (19)\\
&= e^{-\frac{3}{2}j\omega}\left(h_0(e^{\frac{3}{2}j\omega} - e^{-\frac{3}{2}j\omega}) + h_1(e^{\frac{1}{2}j\omega} - e^{-\frac{1}{2}j\omega})\right) && (20)\\
&= e^{-\frac{3}{2}j\omega}\left(2jh_0\sin\left(\tfrac{3}{2}\omega\right) + 2jh_1\sin\left(\tfrac{1}{2}\omega\right)\right) && (21)\\
&= e^{-\frac{3}{2}j\omega}j\left(2h_0\sin\left(\tfrac{3}{2}\omega\right) + 2h_1\sin\left(\tfrac{1}{2}\omega\right)\right) && (22)\\
&= e^{-\frac{3}{2}j\omega}e^{j\frac{\pi}{2}}\left(2h_0\sin\left(\tfrac{3}{2}\omega\right) + 2h_1\sin\left(\tfrac{1}{2}\omega\right)\right) && (23)\\
&= A(\omega)e^{j\theta(\omega)} && (24)
\end{aligned}$$

where

$$\theta(\omega) = -\frac{3}{2}\omega + \frac{\pi}{2}, \qquad A(\omega) = 2h_0\sin\left(\tfrac{3}{2}\omega\right) + 2h_1\sin\left(\tfrac{1}{2}\omega\right).$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\, e^{j\theta(\omega)}$$

In general, for a Type IV FIR filters of length N:

$$H^f(\omega) = A(\omega)\, e^{j\theta(\omega)}$$

$$\boxed{A(\omega) = 2\sum_{n=0}^{\frac{N}{2}-1} h(n)\sin\left((M-n)\omega\right)}$$

$$\boxed{\theta(\omega) = -M\omega + \frac{\pi}{2}}$$

$$\boxed{M = \frac{N-1}{2}.}$$

Video Content / Details of website for further learning (if any): https://www.youtube.com/watch?v=KVOkWcknvc4
Important Books/Journals for further learning including the page nos.: http://eeweb.poly.edu/iselesni/EL713/zoom/linphase.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

LECTURE HANDOUTS

L-22

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Fourier Series

Introduction :

- Fourier series is, in some way a combination of the Fourier sine and Fourier cosine series.
- The Fourier sine/cosine series are not the series will actually converge to f(x) or not at this point.
- The process of deriving the weights that describe a given function is a form of Fourier analysis. For functions on unbounded intervals, the analysis and synthesis analogies are Fourier transform and inverse transform.

Prerequisite knowledge for Complete understanding and learning of Topic:

A Fourier series is an expansion of a periodic function f(x) in terms of an infinite sum of sines and cosines. Fourier series make use of the orthogonality relationships of the sine and cosine functions. The computation and study of Fourier series is known as harmonic analysis and is extremely useful as a way to break up an arbitrary periodic function into a set of simple terms that can be plugged in, solved individually, and then recombined to obtain the solution to the original problem or an approximation to it to whatever accuracy is desired or practical.The mathematical expression

$$A_0 + \sum_{n=1}^{\infty} (A_n \cos(nx) + B_n \sin(nx)).$$

is called fourier series.

Detailed content of the Lecture:

- One of the successful methods, for the design of FIR filters is based on the application of the Fourier series.
- In this method, it is observed that the frequency of an FIR filter is a periodic function of frequency with a period equal to the sampling frequency and consequently, it can be expressed in terms of the Fourier series.
- The Fourier series by itself does not lead to satisfactory results but by using the Fourier series in conjunction with a special class of functions known as window functions, good results can be obtained.
- Approximation obtained by this method is suboptimal but the amounts of design effort and computation required are relatively insignificant**.**

The frequency response of an FIR digital filter can be represented by the Fourier series as,

$$H(\omega)=\sum_{n=-\infty}^{\infty} h(n)e^{-j\omega n}$$

where the Fourier coefficients h(n) are the desired impulse response sequence of the filter, which can be determined from

$$h(n)=\frac{1}{2\pi}\int_{0}^{2\pi} H(\omega)\, e^{+j\omega n}\partial\omega\ .$$

If we substitute e)“ = z, we obtain the transfer function ofthe digital filter, that is,

$$H(z)=\sum_{n=-\infty}^{\infty} h(n)\, z^{-n}.$$

Therefore, if an expression is available for the frequency response, a transfer function can be obtained, which happens to be a non causal and of infinite order. A finite order transfer 35 function can be obtained by truncating the Fourier series. This can be accomplished by letting,

$$h(n)=0 \text{ for} \left| n \right| > \frac{(N-1)}{2}$$

$$H(z)=h(0)+\sum_{n=1}^{\frac{(N-1)}{2}} \left[h(-n)z^{n}+h(n)z^{-n}\right].$$

- This modification does not change the amplitude response of the filter, however the abrupt truncation of the Fourier series results in oscillations in the passband and stopband.
- These oscillations are due to slow convergence of the Fourier series, particularly near points of discontinuity. This effect is known as the Gibbs phenomenon

The amplitude of Gibbs oscillations can be reduced using discrete window functions.

A window function, represented by w(n) has the following time-domain properties:

1. w(n) = 0 for | n j> (N-1)/2
2. It is symmetrical, i.e., w(-n) = w(n).

The application of a window function consists of multiplying the impulse response obtained by applying the Fourier series by the window function to obtain a modified impulse response as

$$hw(n) = w(n)\, h(n)$$

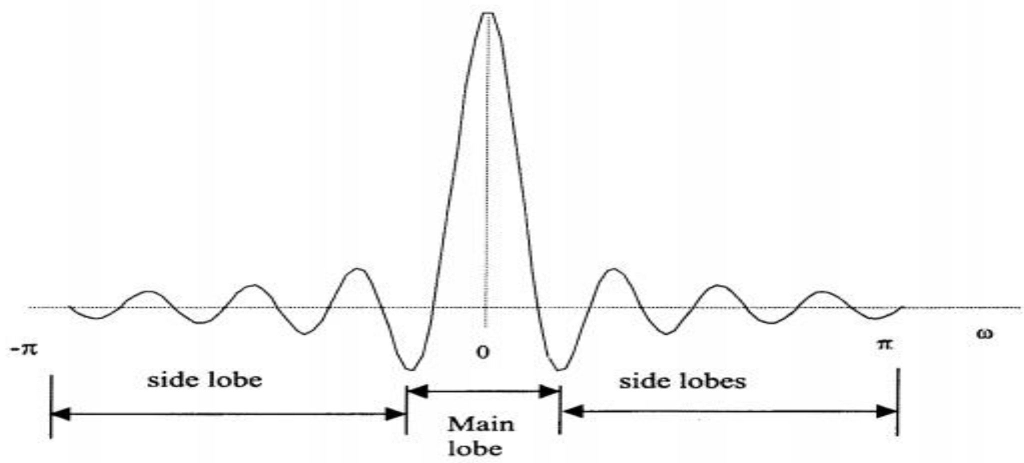


Fig. 2.2 Spectrum of typical window function

Since the window function is of finite duration, a finite-order transfer function given by

$$H_w(z) = \sum_{n=-\frac{(N-1)}{2}}^{\frac{(N-1)}{2}} w(n)\, h(n)\, z^{-n}$$

Through the application of the complex convolution, the frequency response of the modified filter can be expressed as,

$$H_w(\omega) = \frac{1}{2\pi} \int_0^{2\pi} H(\Omega)\ w(e^{j(\omega-\Omega)})\, \partial\Omega$$

The application of the window function has two effects on the amplitude response of the filter. First, the amplitudes of Gibbs' oscillations in the pass bands and stop bands are directly related to the ripple ratio of the window. Second, transition bands are introduced between pass bands and stop bands whose width is directly related to the main-lobe width of the window.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=nDdX11EkOJ0

Important Books/Journals for further learning including the page nos.:

https://shodhganga.inflibnet.ac.in/bitstream/10603/95357/9/09_chapter%202.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-23

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique

Introduction :

- There are essentially three well-known methods for FIR filter design namely:
  (1) The window method
  (2) The frequency sampling technique
  (3) Optimal filter design methods

<u>The Window Method:</u>

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

- The window method for digital filter design is fast, convenient, and robust, but generally suboptimal.
- It is easily understood in terms of the convolution theorem for Fourier transforms, making it instructive to study after the Fourier theorems and windows for spectrum analysis.
- It can be effectively combined with the frequency sampling method.
- The window method consists of simply ``windowing'' a theoretically ideal filter impulse

response $h(n)$ by some suitably chosen window function $w(n)$,

$$h_w(n) = w(n) \cdot h(n), \quad n \in \mathcal{Z}.$$

- Window functions are always *time* limited. This means there is always a finite integer N_w such that $w(n) = 0$ for all $|n| > N_w$.

The final windowed impulse response $h_w(n) = w(n) \cdot h(n)$ is thus always time-limited

Detailed content of the Lecture:

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

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

The unit sample response hd(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 hd(n) to length M-1 is same as multiplying hd(n) by the rectangular window defined as

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

Thus the unit sample response of the FIR filter becomes

$$\begin{aligned} h(n) &= h_d(n)\, w(n) \\ &= h_d(n) \qquad 0 \le n \le M\text{-}1 \\ &= 0 \qquad\quad \text{otherwise} \end{aligned}$$

The multiplication of the window function w(n) with hd(n) is equivalent to convolution of Hd(w) with W(w), where W(w) is the frequency domain representation of the window function

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

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

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

The frequency response can also be obtained using the following relation

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

But direct truncation of hd(n) to M terms to obtain h(n) leads to the Gibbs phenomenon effect which manifests itself as a fixed percentage overshoot and ripple before and after an approximated discontinuity in the frequency response due to the non-uniform convergence of the fourier series at a discontinuity. Thus the frequency response obtained by using (8) contains 3 ripples in the frequency domain. In order to reduce the ripples, instead of multiplying hd(n) with a rectangular window w(n), hd(n) is multiplied with a window function.

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

(i) A major effect is that discontinuities in H(w) become transition bands between values on either side of the discontinuity.

(ii) 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).

(iii) Since the filter frequency response is obtained via a convolution relation, it is clear that the resulting filters are never optimal in any sense.

(iv) As M (the length of the window function) increases, the main lobe width of W(w) is reduced which reduces the width of the transition band, but this also introduces more ripple in the frequency response.

(v) The window function eliminates the ringing effects at the band edge and does result in lower side lobes at the expense of an increase in the width of the transition band of the filter.

Video Content / Details of website for further learning (if any):
https://www.youtube.com/watch?v=KVOkWcknvc4

Important Books/Journals for further learning including the page nos.:
https://www.ee.iitb.ac.in/~esgroup/es_mtech02_sem/es02_sem_rep_arojit.pdf
https://www.allaboutcircuits.com/technical-articles/finite-impulse-response-filter-design-by-windowing-part-i-concepts-and-rect/

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-24

ECE

III/V

Course Name with Code : **16ECD09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **FIR FILTER DESIGN** **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique-Rectangular window

Introduction :

- There are essentially three well-known methods for FIR filter design namely:
  (1) The window method
  (2) The frequency sampling technique
  (3) Optimal filter design methods

<u>The Window Method:</u>

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

- A window is a finite array consisting of coefficients selected to satisfy the desirable requirements.
- While designing digital FIR filter using window function it is necessary to specify a window function to be used and the filter order according to the required specifications (selectivity and stop band attenuation).
- These two requirements are interrelated .

- Window Technique implicates a function called window Function.
- It is also known as Tapering Function.

Detailed content of the Lecture:

Rectangular Window

The rectangular window (sometimes known as the Boxer or Dirichlet window) is the simplest window, equivalent to replacing all but N values of a data sequence by zeros, making it appear as though the waveform suddenly turns on and off.

$$w_R(n) \triangleq \begin{cases} 1, & -\frac{M-1}{2} \leq n \leq \frac{M-1}{2} \\ 0, & \text{otherwise} \end{cases}$$

where M is the window length in samples (assumed odd for now). A plot of the rectangular window appears in Fig.3.1 for length $M = 21$. It is sometimes convenient to define windows so that their dc gain is 1, in which case we would multiply the definition above by $1/M$

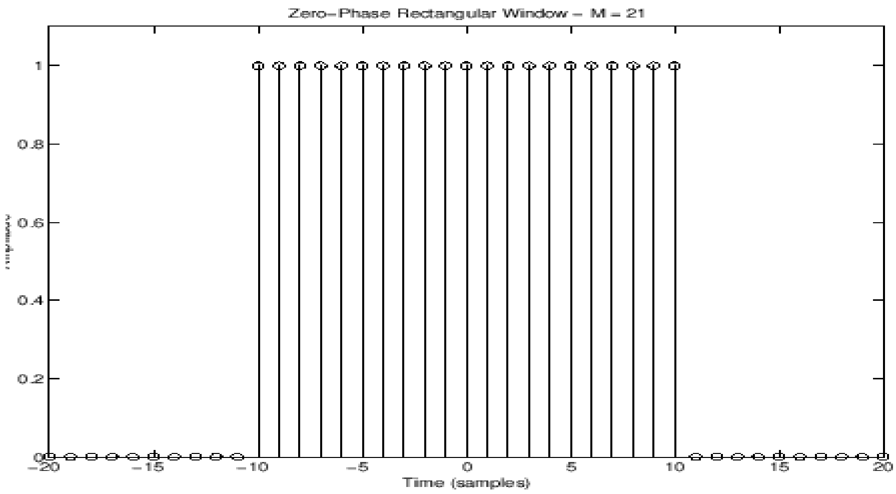


Figure 3.1: The rectangular window.

$$\begin{aligned} W_R(\omega) &= \text{DTFT}_\omega(w_R) \triangleq \sum_{n=-\infty}^{\infty} w_R(n)e^{-j\omega n}, \quad \omega \in [-\pi, \pi) \\ &= \sum_{n=-\frac{M-1}{2}}^{\frac{M-1}{2}} e^{-j\omega n} = \frac{e^{j\omega\frac{M-1}{2}} - e^{-j\omega\frac{M+1}{2}}}{1 - e^{-j\omega}} \end{aligned}$$

$$W_R(\omega) = \frac{e^{-j\omega\frac{1}{2}}}{e^{-j\omega\frac{1}{2}}}\left[\frac{e^{j\omega\frac{M}{2}} - e^{-j\omega\frac{M}{2}}}{e^{j\omega\frac{1}{2}} - e^{-j\omega\frac{1}{2}}}\right]$$

$$= \frac{\sin\left(M\frac{\omega}{2}\right)}{\sin\left(\frac{\omega}{2}\right)} \triangleq M \cdot \text{asinc}_M(\omega)$$

$$\text{asinc}_M(\omega) \triangleq \frac{\sin(M\omega/2)}{M \cdot \sin(\omega/2)}$$

also called the Dirichlet function or periodic sinc function. This (real) result is for the zero-centered rectangular window. For the causal case, a linear phase term appears:

$$W_R^c(\omega) = e^{-j\frac{M-1}{2}\omega} \cdot M \cdot \text{asinc}_M(\omega)$$

The term ``aliased sinc function" refers to the fact that it may be simply obtained by sampling the length- τ continuous-time rectangular window, which has Fourier transform sinc $(f\tau) \triangleq \sin(\pi f\tau)/(\pi f\tau)$ (given amplitude $1/\tau$ in the time domain). Sampling at intervals of T seconds in the time domain corresponds to aliasing in the frequency domain over the interval $[0, 1/T]$ Hz, and by direct derivation. The window duration increases continuously in the time domain: the magnitude spectrum can only change in discrete jumps as new samples are included, even though it is continuously parametrized in τ .As the sampling rate goes to infinity, the aliased sinc function therefore approaches the sinc function.

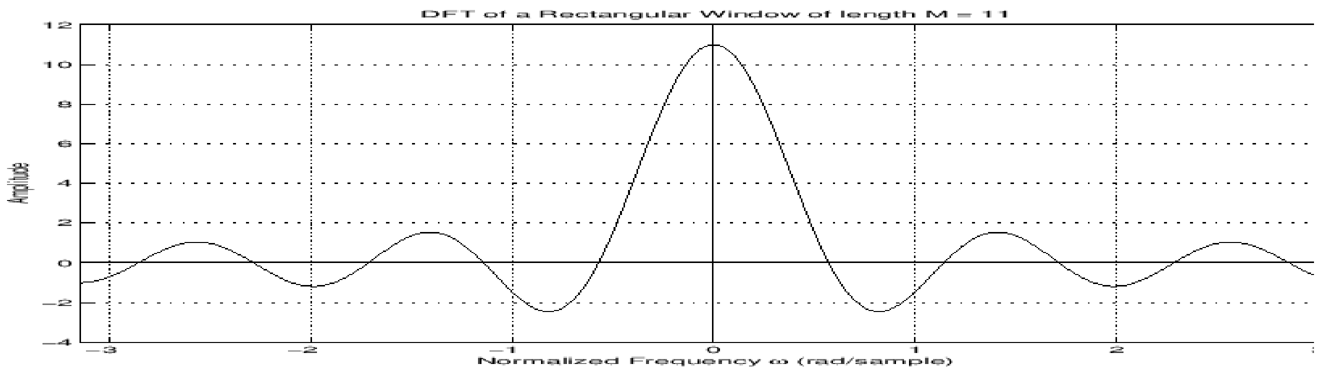


Figure: Fourier transform of the rectangular window.

The phase of rectangular-window transform $W_R(\omega)$ is zero for $|\omega| < 2\pi/M$, which is the width of the main lobe. This is why zero-centered windows are often called zero-phase windows; while the phase actually alternates between 0 and π radians, the π values occur only within side-lobes which are routinely neglected.

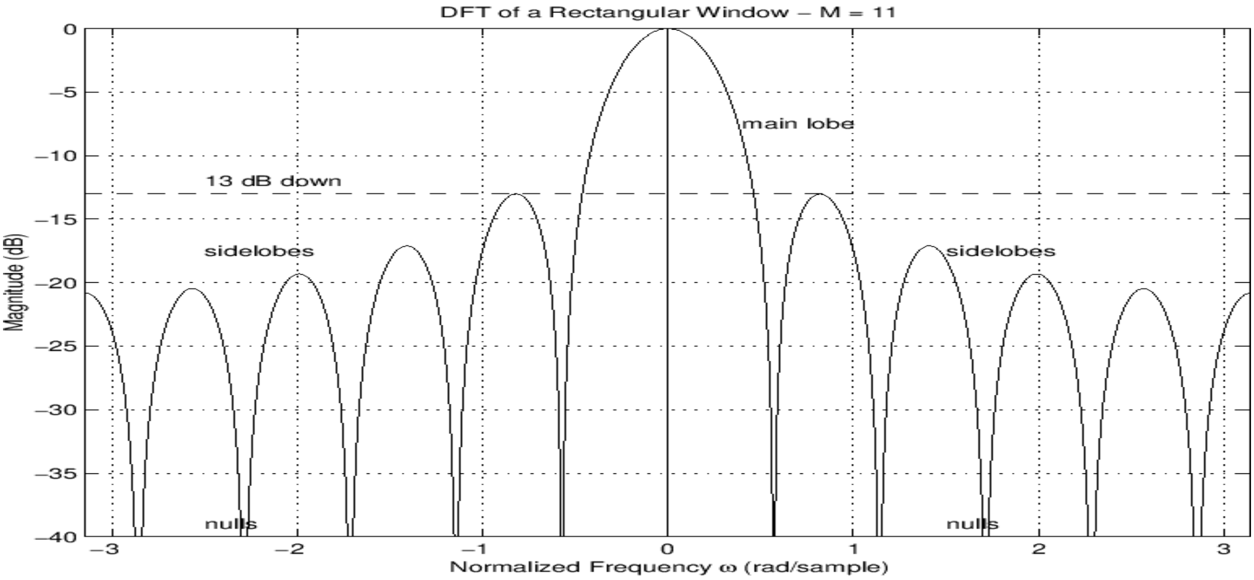


Figure: Magnitude (dB) of the rectangular-window transform

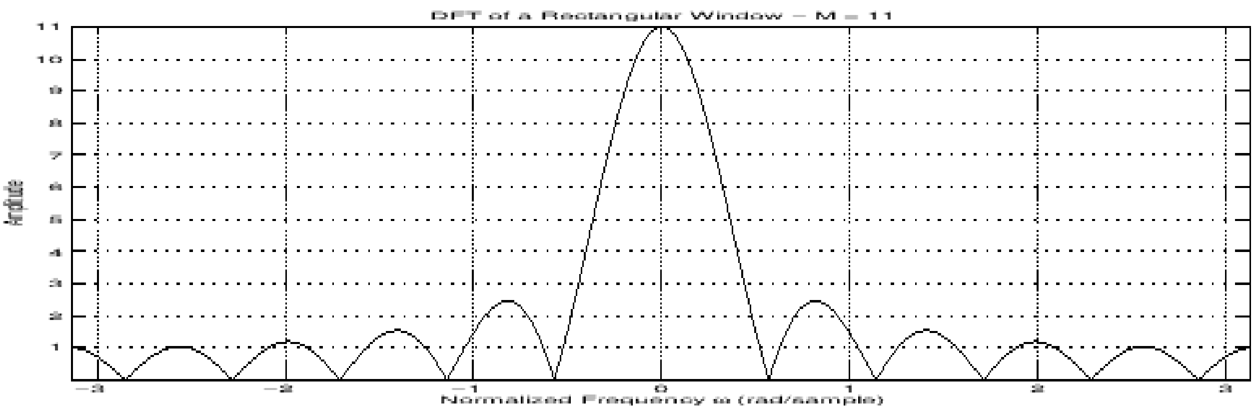


Figure: Magnitude of the rectangular-window Fourier transform.

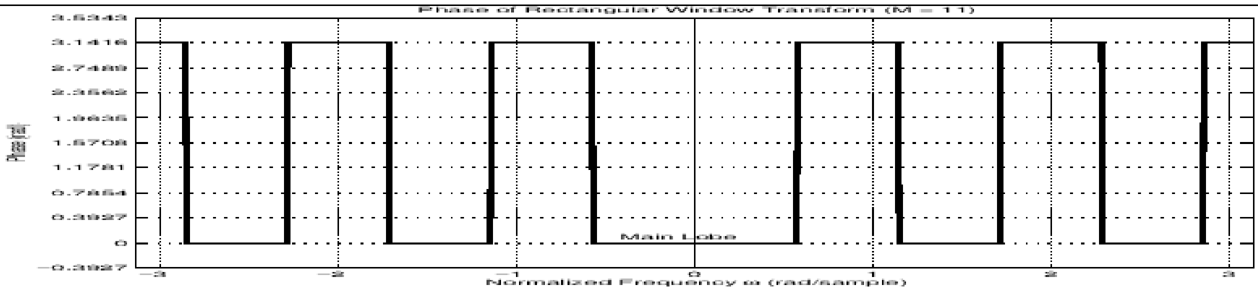


Figure: Phase of the rectangular-window Fourier transform.

Video Content / Details of website for further learning (if any):
https://www.youtube.com/watch?v=s1-RMohyJ1A

Important Books/Journals for further learning including the page nos.:
https://ccrma.stanford.edu/~jos/sasp/Rectangular_Window.html
https://www.dsprelated.com/freebooks/sasp/Rectangular_Window.html

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)

Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-25

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : DESIGN OF FIR FILTER **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique-Hamming window

Introduction :

- There are essentially three well-known methods for FIR filter design namely:
  (1) The window method
  (2) The frequency sampling technique
  (3) Optimal filter design methods

The Window Method:

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

- A window is a finite array consisting of coefficients selected to satisfy the desirable requirements.
- While designing digital FIR filter using window function it is necessary to specify a window function to be used and the filter order according to the required specifications (selectivity and stop band attenuation).
- These two requirements are interrelated .
- Window Technique implicates a function called window Function.

- It is also known as Tapering Function.

Detailed content of the Lecture:

Hamming Window

The Hamming window is determined by choosing α (with $\beta \triangleq (1-\alpha)/2$) to cancel the largest side lobe. Doing this results in the values

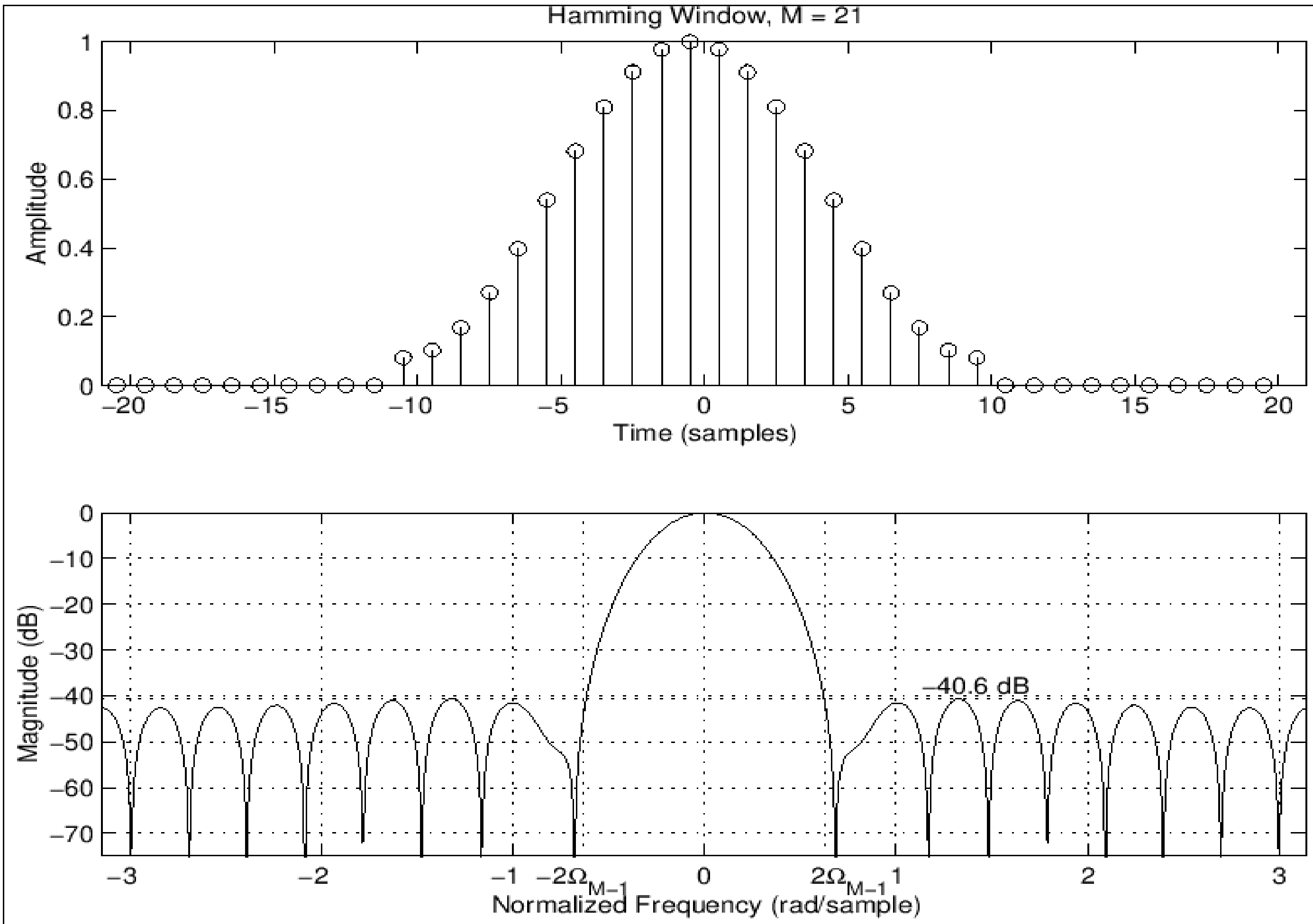
$$\begin{aligned} \alpha &= \frac{25}{46} \approx 0.54 \\ \beta &= \frac{1-\alpha}{2} \approx 0.23. \end{aligned}$$

The peak side-lobe level is approximately -42.76 dB for the Hamming window . It happens that this choice is very close to that which minimizes peak side-lobe level (down to -43.19 dB--the lowest possible within the generalized Hamming family).

$$\alpha = 0.53836\ldots$$

Since rounding the optimal α to two significant digits gives 0.54 , the Hamming window can be considered the ``Chebyshev Generalized Hamming Window" (approximately). Chebyshev-type designs normally exhibit equiripple error behavior, because the worst-case error (side-lobe level in this case) is minimized. Generalized Hamming windows can have a step discontinuity at their endpoints, but no impulsive points.

- The Hamming window and its DTFT magnitude are shown in Figure. The Hamming window is also one period of a raised cosine.
- However, the cosine is raised so high that its negative peaks are above zero, and the window has a discontinuity in amplitude at its endpoints (stepping discontinuously from 0.08 to 0).
- This is 10 dB better than the Hann case of Figure and 28 dB better than the rectangular window.
- The main lobe is approximately wide, as is the case for all members of the generalized Hamming family



For the Hamming window, the side-lobes nearest the main lobe have been strongly shaped by the optimization. As a result, the nearly -6 dB per octave roll-off occurs only over an interior interval of the spectrum, well between the main lobe and half the sampling rate. The optimized side-lobes nearest the main lobe occupy a smaller frequency interval about the main lobe.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=4e_ffrF6HT4

Important Books/Journals for further learning including the page nos.:

https://ccrma.stanford.edu/~jos/sasp/Hamming_Window.html

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

LECTURE HANDOUTS

L-26

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Filter design using Windowing technique-Hanning window

Introduction :

- There are essentially three well-known methods for FIR filter design namely:
  (1) The window method
  (2) The frequency sampling technique
  (3) Optimal filter design methods

The Window Method:

- In this method the desired frequency response specification Hd(w), corresponding unit sample response hd(n) is determined using the following relation

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

where

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

Prerequisite knowledge for Complete understanding and learning of Topic:

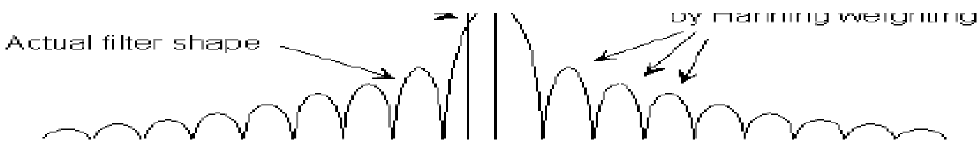
- A window is a finite array consisting of coefficients selected to satisfy the desirable requirements.
- While designing digital FIR filter using window function it is necessary to specify a window function to be used and the filter order according to the required specifications (selectivity and stop band attenuation).

- These two requirements are interrelated .
- Window Technique implicates a function called window Function.
- It is also known as Tapering Function.

Detailed content of the Lecture:

<u>Hanning Window</u>

- Hanning window is the shape of one cycle of a cosine wave with 1 added to it so it is always positive.
- The sampled signal values are multiplied by the Hanning function.
- The ends of the time record are forced to zero regardless of what the input signal is doing.
- While the Hanning window does a good job of forcing the ends to zero, it also adds distortion to the wave form being analyzed in the form of amplitude modulation; i.e., the variation in amplitude of the signal over the time record.
- Amplitude Modulation in a wave form results in sidebands in its spectrum, and in the case of the Hanning window, these sidebands, or side lobes as they are called, effectively reduce the frequency resolution of the analyzer by 50%.
- It is as if the analyzer frequency "lines" are made wider.
- The highest-level side lobes are about 32 dB down from the main lobe.



- The Hanning window should always be used with continuous signals, but must never be used with transients. The reason is that the window shape will distort the shape of the transient, and the frequency and phase content of a transient is intimately connected with its shape.
- The measured level will also be greatly distorted. Even if the transient were in the center of the Hanning window, the measured level would be twice as great as the actual level because of the amplitude correction the analyzer applies when using the Hanning weighting.
- A Hanning weighted signal actually is only half there, the other half of it having been removed by the windowing. This is not a problem with a perfectly smooth and continuous signal like a sinusoid, but most signals we want to analyze, such as machine vibration signatures are not perfectly smooth.
- If a small change occurs in the signal near the beginning or end of the time record, it will either be analyzed at a much lower level than its true level, or it may be missed altogether.

- For this reason, it is a good idea to employ overlap processing. To do this, two time buffers are required in the analyzer.
- For 50% overlap, the sequence of events is as follows:
- When the first buffer is half full, i.e., it contains half the samples of a time record, the second buffer is connected to the data stream and also begins to collect samples. As soon as the first buffer is full, the FFT is calculated, and the buffer begins to take data again.
- When the second buffer is filled, the FFT is again calculated on its contents, and the result sent to the spectrum-averaging buffer. This process continues on until the desired number of averages is collected.

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=s3sOhj7DaFg

Important Books/Journals for further learning including the page nos.:

http://www.azimadli.com/vibman/thehanningwindow.htm

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-27

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **DESIGN OF FIR FILTER** **Date of Lecture :**

Topic of Lecture: Frequency sampling method

Introduction :

- The frequency-sampling method for FIR filter design is perhaps the simplest and most direct technique imaginable when a desired frequency response has been specified.
- It consists simply of uniformly sampling the desired frequency response, and performing an inverse DFT to obtain the corresponding (finite) impulse response.
- The results are not optimal, however, because the response generally deviates from what is desired between the samples.
- When the desired frequency-response is undersampled, which is typical, the resulting impulse response will be time aliased to some extent.

Prerequisite knowledge for Complete understanding and learning of Topic:

- The main idea of the frequency sampling design method is that a desired frequency response can be approximated by sampling it of N evenly spaced points and then obtaining an interpolated frequency response that passes through the frequency samples.
- For filters with reasonably smooth frequency responses,the interpolation error is generally small.
- In the case of band select filters, where the desired frequency response changes radically across bands, the frequency samples which occur in transition bands are made to be unspecified variables whose values are chosen by an optimization algorithm which minimizes some function of the

approximation error of the filter finally, it was shown that there were two distinct types of frequency sampling filters, depending on where the initial frequency sample.

Detailed content of the Lecture:

The frequency sampling method allows us to design recursive and nonrecursive FIR filters for both standard frequency selective and filters with arbitrary frequency response.

No recursive frequency sampling filters :

The problem of FIR filter design is to find a finite–length impulse response h (n) that corresponds to desired frequency response. In this method h (n) can be determined by uniformly sampling, the desired frequency response HD (ω) at the N points and finding its inverse DFT of the frequency samples

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

where H (k), k = 0, 1, 2,……., N-1, are samples of the HD (ω) . For linear phase filters, with positive symmetrical impulse response, we can write

$$h(n)=\frac{1}{N}\left[\sum_{k=1}^{(N/2)-1}2|H(k)|\cos\left[\frac{2\pi k(n-\alpha)}{N}\right]+H(0)\right]$$

where α = (N-1)/2. For N odd, the upper limit in the summation is (N - 1)/2, to obtain a good approximation to the desired frequency

Recursive frequency sampling filter :

In recursive frequency sampling method the DFT samples H (k) for an FIR sequence can be regarded as samples of the filters z– transform, evaluated at N points equally spaced around the unit circle.

$$H(k)=H(z)\big|_{z=e^{j(2\pi/N)k}}$$

thus the z–transform of an FIR filter can easily be expressed in terms of its DFT coefficients,

$$H(z)=\sum_{n=0}^{N-1}h(n)z^{-n}=\sum_{n=0}^{N-1}\left[\frac{1}{N}\sum_{k=0}^{N-1}H(k)e^{j(2\pi/N)nk}\right]z^{-n}$$

$$= \sum_{k=0}^{N-1} \frac{H(k)}{N} \sum_{n=0}^{N-1} \left[e^{j(2\pi/N)k} z^{-1} \right]^n$$

$$= \sum_{k=0}^{N-1} \frac{H(k)}{N} \frac{(1 - e^{j2\pi k} z^{N})}{(1 - e^{j(2\pi/N)k} z^{-1})}$$

by butting $e^{j2\pi k} = 1$, Equation (13) reduces to

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

This is the desired result

Video Content / Details of website for further learning (if any):

https://www.youtube.com/watch?v=O0DbpZqhPSg

Important Books/Journals for further learning including the page nos.:

https://bulletin.zu.edu.ly/issue_n15_3/Contents/E_04.pdf

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

L-37

LECTURE HANDOUTS

ECE **II/IV**

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./EC**

Unit : **MULTIRATE AND DIGITAL SIGNAL PROCESSORS**

Date of Lecture :

Topic of Lecture: **Multirate signal processing**

Introduction :

The sampling rate of a signal is changed in order to increase the efficiency of various signal processing operations. Decimation, or down-sampling, reduces the sampling rate, whereas expansion, or up-sampling, followed by interpolation increases the sampling rate.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Signal Processing

Detailed content of the Lecture:

Up-sampling

- Increasing the sampling frequency, before D/A conversion in order to relax the requirements of the analog lowpass antialiasing filter.
- This technique is used in audio CD, where the sampling frequency 44.1 kHz is increased fourfold to 176.4 kHz before D/A conversion.

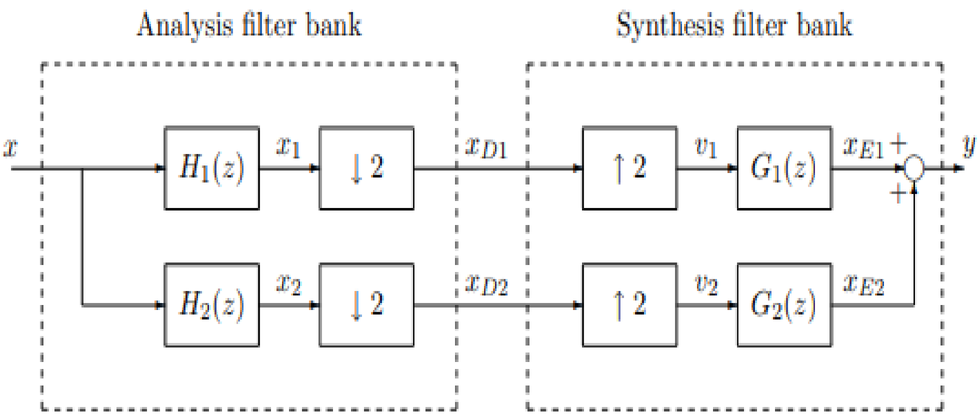
Decomposition

- Decomposition of a signal into M components containing various frequency bands. If the original signal is sampled at the sampling frequency fs (with a frequency band of width fs/2, or half the sampling frequency), every component then contains a frequency band of width 1 2 fs/M only, and can be represented using the sampling rate fs/M.
- This allows for efficient parallel signal processing with processors operating at lower sampling rates. The technique is also applied to data compression in subband coding, for example in speech processing, where the various frequency band components are represented with different word lengths.
- In the implementation of high-performance filtering operations, where a very narrow transition band is required. The requirement of narrow transition bands leads to very high filter orders.
- However, by decomposing the signal into a number of subbands containing the passband, stopband and transition bands, each component can be processed at a lower rate, and the transition band will be less narrow. Hence the required filter complexity may be reduced significantly.

Subband decomposition

- In order to present the basic techniques involved in decomposing a signal into subband components, let's consider a simple case where a signal is decomposed into two components: a low-frequency component and a high-frequency component.
- The purpose of the filters H1 and H2 is to extract the low- and high-frequency components of the signal x. For perfect signal decomposition, H1 should be an ideal low-pass filter with the passband [0, π/2], and H2 should be an ideal high-pass filter with the passband [π/2, π], cf.

$$X_{D1}(\omega) = 1/2\ X(\omega/2),\ 0 \le \omega < \pi$$
$$X_{D2}(\omega) = 1/2\ X(\pi - \omega/2)*,\ 0 \le \omega < \pi$$



- Real filters characteristics resemble more the general form. It follows that it is not possible to separate the frequency bands exactly, but instead either some aliasing between the frequency bands is unavoidable, or, if the frequencies at the band edges are attenuated completely, some frequencies are lost altogether.
- The standard solution to the aliasing problem is to design the filters H1 and H2 in such a way that despite aliasing, it is still possible to reconstruct the original signal from the components. This can be achieved with quadrature mirror filters.

Important Video/Website Content:
http://users.abo.fi/htoivone/courses/sbappl/asp_chapter2.pdf

Important Books/Journals for further learning including the page nos.:

1. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(839-843)**

Course Teacher

Verified by HOD

L-38

LECTURE HANDOUTS

ECE **II/IV**

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **MULTIRATE AND DIGITAL SIGNAL PROCESSORS**

Date of Lecture :

Topic of Lecture: **Decimation**

Introduction :

- **Decimation** is the process of reducing the sampling rate of a signal.
- The term **downsampling** usually refers to one step of the process, but sometimes the terms are used interchangeably. Complementary to upsampling, which increases sampling rate, decimation is a specific case of sample rate conversion in a multi-rate digital signal processing system. A system component that performs decimation is called a decimator

Prerequisite knowledge for Complete understanding and learning of Topic:

- Down Sampling
- Up Samling

Detailed content of the Lecture:

- When decimation is performed on a sequence of samples of a *signal* or other continuous function, it produces an approximation of the sequence that would have been obtained by sampling the signal at a lower rate (or density, as in the case of a photograph).
- The *decimation factor* is usually an integer or a rational fraction greater than one. This factor multiplies the sampling interval or, equivalently, divides the sampling rate. For example, if compact disc audio at 44,100 samples/second is decimated by a factor of 5/4, the resulting sample rate is 35,280.

Decimation by an integer factor

- Decimation by an integer factor, *M*, can be explained as a two-step process, with an equivalent implementation that is more efficient:
- Reduce high-frequency signal components with a digital lowpass filter.
- Downsample the filtered signal by *M*; that is, keep only every M^{th} sample.
- Downsampling alone causes high-frequency signal components to be misinterpreted by subsequent users of the data, which is a form of distortion called aliasing. The first step, if necessary, is to suppress aliasing to an acceptable level. In this application, the filter is called an anti-aliasing filter, and its design is discussed below. Also see undersampling for information about downsampling bandpass functions and signals.
- When the anti-aliasing filter is an IIR design, it relies on feedback from output to input, prior to the downsampling step.

- With FIR filtering, it is an easy matter to compute only every M^{th} output. The calculation performed by a decimating FIR filter for the n^{th} output sample is a dot product:

$$Y(n) = \sum x[nM-k].h(k)$$

- where the h[k] sequence is the impulse response, and K is its length. x[k] represents the input sequence being downsampled. In a general purpose processor, after computing $y[n]$, the easiest way to compute $y[n+1]$ is to advance the starting index in the x[k] array by M, and recompute the dot product. In the case M=2, h[k] can be designed as a half-band filter, where almost half of the coefficients are zero and need not be included in the dot products.
- Impulse response coefficients taken at intervals of M form a subsequence, and there are M such subsequences (phases) multiplexed together. The dot product is the sum of the dot products of each subsequence with the corresponding samples of the x[k] sequence.
- Furthermore, because of downsampling by M, the stream of x[k] samples involved in any one of the M dot products is never involved in the other dot products. Thus M low-order FIR filters are each filtering one of M multiplexed *phases* of the input stream, and the M outputs are being summed.
- In other words, the input stream is demultiplexed and sent through a bank of M filters whose outputs are summed. When implemented that way, it is called a polyphase filter.
- For completeness, we now mention that a possible, but unlikely, implementation of each phase is to replace the coefficients of the other phases with zeros in a copy of the h[k] array, process the original x[k] sequence at the input rate, and decimate the output by a factor of M.
- The equivalence of this inefficient method and the implementation described above is known as the *first Noble identity*

Anti-aliasing filter

- The requirements of the anti-aliasing filter can be deduced from any of the three pairs of graphs.
- Note that all three pairs are identical, except for the units of the abscissa variables. The upper graph of each pair is an example of the periodic frequency distribution of a sampled function, $x(t)$, with Fourier transform, $X(f)$.
- The lower graph is the new distribution that results when $x(t)$ is sampled three times slower, or (equivalently) when the original sample sequence is decimated by a factor of M=3.
-

Important Video/Website Content:
https://wiki2.org/en/Decimation_(signal_processing)

Important Books/Journals for further learning including the page nos.:

2. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(783-787)**

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

L-39

LECTURE HANDOUTS

ECE **II/IV**

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : MULTIRATE AND DIGITAL SIGNAL PROCESSORS

Date of Lecture :

Topic of Lecture: **Interpolation**

Introduction :

- Interpolation is a type of estimation, a method of constructing new data points within the range of a discrete set of known data points.
- It is often required to interpolate, i.e., estimate the value of that function for an intermediate value of the independent variable.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Down Sampling
- Up Samling

Detailed content of the Lecture:

Linear interpolation

- Generally, linear interpolation takes two data points, say (x_a,y_a) and (x_b,y_b), and the interpolant is given by:

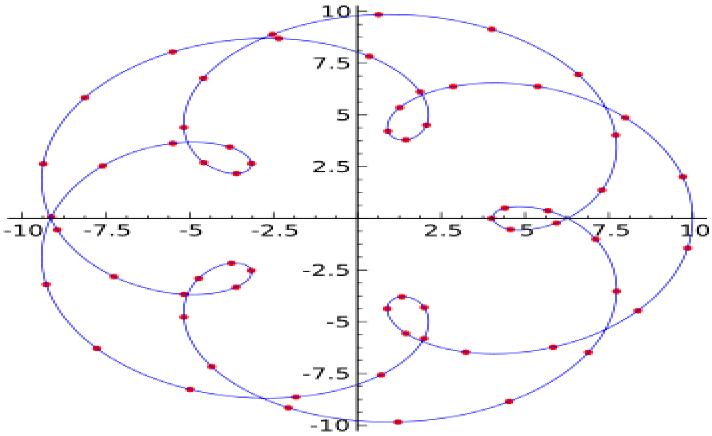
$$y = y_a + (y_b - y_a)\frac{x - x_a}{x_b - x_a} \text{ at the point } (x, y)$$

$$\frac{y - y_a}{y_b - y_a} = \frac{x - x_a}{x_b - x_a}$$

$$\frac{y - y_a}{x - x_a} = \frac{y_b - y_a}{x_b - x_a}$$

- Linear interpolation is quick and easy, but it is not very precise. Another disadvantage is that the interpolant is not differentiable at the point x_k.
- The following error estimate shows that linear interpolation is not very precise. Denote the function which we want to interpolate by g, and suppose that x lies between x_a and x_b and that g is twice continuously differentiable.

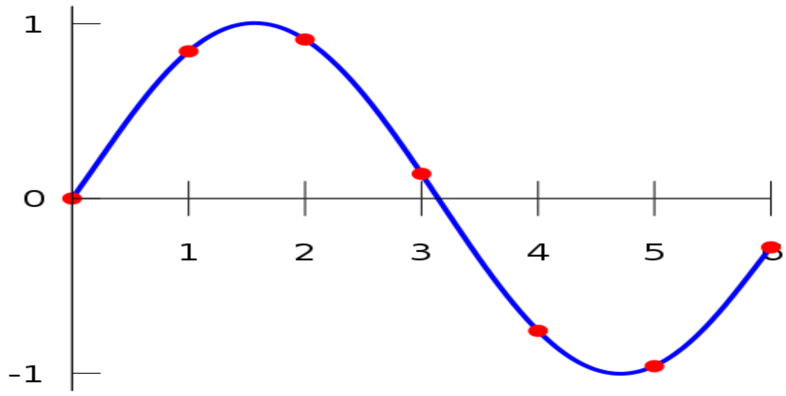
- Then the linear interpolation error is



Interpolation of Finite Set Points

Polynomial interpolation

- Polynomial interpolation is a generalization of linear interpolation. Note that the linear interpolant is a linear function. We now replace this interpolant with a polynomial of higher degree.



- Polynomial interpolation also has some disadvantages. Calculating the interpolating polynomial is computationally expensive (see computational complexity) compared to linear interpolation. Furthermore, polynomial interpolation may exhibit oscillatory artifacts, especially at the end points.
- Polynomial interpolation can estimate local maxima and minima that are outside the range of the samples, unlike linear interpolation.

Important Video/Website Content:
https://wiki2.org/en/Interpolation

Important Books/Journals for further learning including the page nos.:

3. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(787)**

Course Teacher

Verified by HOD

L-40

LECTURE HANDOUTS

ECE **II/IV**

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **MULTIRATE AND DIGITAL SIGNAL PROCESSORS**

Date of Lecture :

Topic of Lecture: Cascading Sample Rate Converters

Introduction :

- Sample-rate conversion is the process of changing the sampling rate of a discrete signal to obtain a new discrete representation of the underlying continuous signal.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Down Sampling
- Up Samling

Detailed content of the Lecture:

- Sample-rate conversion is the process of changing the sampling rate of a discrete signal to obtain a new discrete representation of the underlying continuous signal. Application areas include image scaling and audio/visual systems, where different sampling rates may be used for engineering, economic, or historical reasons.

1. Conceptual approaches to sample-rate conversion include: converting to an analog continuous signal, then re-sampling at the new rate, or calculating the values of the new samples directly from the old samples.
2. If the ratio of the two sample rates is (or can be approximated by) a fixed rational number L/M: generate an intermediate signal by inserting $L - 1$ 0s between each of the original samples. Low-pass filter this signal at half of the lower of the two rates. Select every M-th sample from the filtered output, to obtain the result.
3. Treat the samples as geometric points and create any needed new points by interpolation. Choosing an interpolation method is a trade-off between implementation complexity and conversion quality (according to application requirements). Commonly used are: ZOH (for film/video frames), cubic (for image processing) and windowed sinc function (for audio).

Anti-aliasing

- Oversampling can make it easier to realize analog anti-aliasing filters. Without oversampling, it is very difficult to implement filters with the sharp cutoff necessary to maximize use of the available bandwidth without exceeding the Nyquist limit.
- By increasing the bandwidth of the sampling system, design constraints for the anti-aliasing filter may be relaxed.

- Once sampled, the signal can be digitally filtered and downsampled to the desired sampling frequency. In modern integrated circuit technology, the digital filter associated with this downsampling are easier to implement than a comparable analog filter required by a non-oversampled system.

Resolution

- In practice, oversampling is implemented in order to reduce cost and improve performance of an analog-to-digital converter (ADC) or digital-to-analog converter (DAC).[1] When oversampling by a factor of N, the dynamic range also increases a factor of N because there are N times as many possible values for the sum.

Noise

- If multiple samples are taken of the same quantity with uncorrelated noise added to each sample, then because, as discussed above, uncorrelated signals combine more weakly than correlated ones, averaging N samples reduces the noise power by a factor of N.
- If, for example, we oversample by a factor of 4, the signal-to-noise ratio in terms of power improves by factor of 4 which corresponds to a factor of 2 improvement in terms of voltage.

Oversampling in reconstruction

- The term oversampling is also used to denote a process used in the reconstruction phase of digital-to-analog conversion, in which an intermediate high sampling rate is used between the digital input and the analogue output.
- Digital interpolation is used to add additional samples between recorded samples, thereby converting the data to a higher sample rate, a form of upsampling.
- When the resulting higher-rate samples are converted to analog, a less complex and less expensive analog reconstruction filter is required. Essentially, this is a way to shift some of the complexity of reconstruction from analog to the digital domain.
- Oversampling in the ADC can achieve some of the same benefits as using a higher sample rate at the DAC.
- In signal processing, undersampling or bandpass sampling is a technique where one samples a bandpass-filtered signal at a sample rate below its Nyquist rate (twice the upper cutoff frequency), but is still able to reconstruct the signal.
- When one undersamples a bandpass signal, the samples are indistinguishable from the samples of a low-frequency alias of the high-frequency signal. Such sampling is also known as bandpass sampling, harmonic sampling, IF sampling, and direct IF-to-digital conversion

Important Books/Journals for further learning including the page nos.:

4. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(837-843)**

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-41

ECE

II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : MULTIRATE AND DIGITAL SIGNAL PROCESSORS

Date of Lecture :

Topic of Lecture: **Efficient Transversal Structure for Decimator**

Introduction :

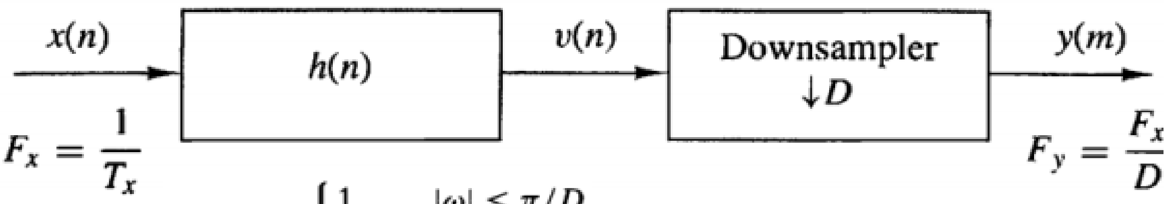
- Decimation and interpolation are the two basic building blocks in the multirate digital signal processing systems. As the linear canonical transform (LCT) has been shown to be a powerful tool for optics and signal processing, it is worthwhile and interesting to analyze the decimation and interpolation in the LCT domain. .

Prerequisite knowledge for Complete understanding and learning of Topic:

- Decimator

Detailed content of the Lecture:

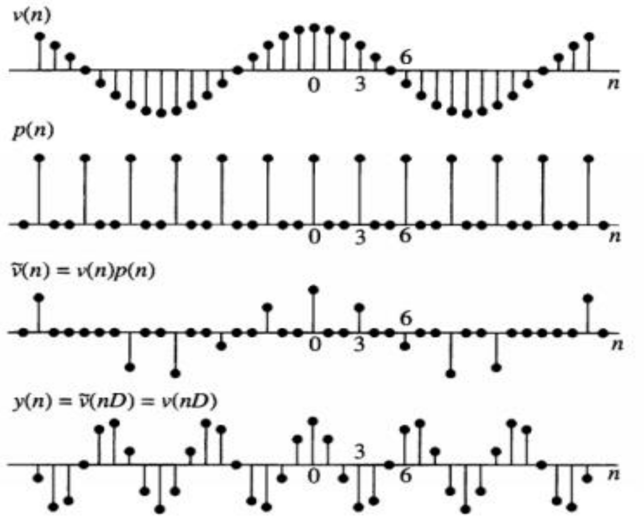
Decimation by a factor D



$$H_D(\omega) = \begin{cases} 1, & |\omega| \le \pi/D \\ 0, & \text{otherwise} \end{cases}$$

- $$v(n) = \sum_{k=0}^{\infty} h(k)x(n-k) \qquad y(m) = v(mD) = \sum_{k=0}^{\infty} h(k)x(mD-k)$$

Decimation by a factor D



$$\tilde{v}(n) = \begin{cases} v(n), & n = 0, \pm D, \pm 2D, \ldots \\ 0, & \text{otherwise} \end{cases}$$

$$\tilde{v}(n) = v(n)p(n)$$

$$p(n) = \frac{1}{D}\sum_{k=0}^{D-1} e^{j2\pi kn/D}$$

$$y(m) = \tilde{v}(mD) = v(mD)p(mD) = v(mD)$$

Decimation by a factor D

$$Y(z) = \sum_{m=-\infty}^{\infty} y(m)z^{-m} = \sum_{m=-\infty}^{\infty} \tilde{v}(mD)z^{-m} \quad \Rightarrow \quad Y(z) = \sum_{m=-\infty}^{\infty} \tilde{v}(m)z^{-m/D}$$

$$\tilde{v}(n) = v(n)p(n)$$

$$p(n) = \frac{1}{D}\sum_{k=0}^{D-1} e^{j2\pi kn/D}$$

$$\Rightarrow$$

$$\begin{aligned} Y(z) &= \sum_{m=-\infty}^{\infty} v(m)\left[\frac{1}{D}\sum_{k=0}^{D-1} e^{j2\pi mk/D}\right] z^{-m/D} \\ &= \frac{1}{D}\sum_{k=0}^{D-1}\sum_{m=-\infty}^{\infty} v(m)(e^{-j2\pi k/D}z^{1/D})^{-m} \\ &= \frac{1}{D}\sum_{k=0}^{D-1} V(e^{-j2\pi k/D}z^{1/D}) \\ &= \frac{1}{D}\sum_{k=0}^{D-1} H_D(e^{-j2\pi k/D}z^{1/D})X(e^{-j2\pi k/D}z^{1/D}) \end{aligned}$$

$$v(n) = \sum_{k=0}^{\infty} h(k)x(n-k)$$

$$H_D(\omega) = \begin{cases} 1, & |\omega| \le \pi/D \\ 0, & \text{otherwise} \end{cases}$$

Decimation by a factor D

Evaluate the Z-transform on unit circle with frequency variable $\omega_y = \frac{2\pi F}{F_y} = 2\pi F T_y$

$$F_y = \frac{F_x}{D} \qquad \omega_x = \frac{2\pi F}{F_x} = 2\pi F T_x \quad \Rightarrow \quad \omega_y = D\omega_x$$

Thus, $0 \le |\omega_x| \le \pi/D$ gets stretched to $0 \le |\omega_y| \le \pi$ by down-sampling

$$Y(z) = \frac{1}{D}\sum_{k=0}^{D-1} H_D(e^{-j2\pi k/D} z^{1/D}) X(e^{-j2\pi k/D} z^{1/D})$$

$$\Downarrow$$

$$Y(\omega_y) = \frac{1}{D}\sum_{k=0}^{D-1} H_D\left(\frac{\omega_y - 2\pi k}{D}\right) X\left(\frac{\omega_y - 2\pi k}{D}\right)$$

If $H_D(\omega)$ is correctly designed, then aliasing is eliminated and

$$Y(\omega_y) = \frac{1}{D} H_D\left(\frac{\omega_y}{D}\right) X\left(\frac{\omega_y}{D}\right) = \frac{1}{D} X\left(\frac{\omega_y}{D}\right) \qquad \text{for } 0 \le |\omega_y| \le \pi$$

Important Books/Journals for further learning including the page nos.:

5. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(784-789)**

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

LECTURE HANDOUTS

L-42

ECE

II/IV

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **MULTIRATE AND DIGITAL SIGNAL PROCESSORS**

Date of Lecture :

Topic of Lecture: **Efficient Transversal Structure for Interpolator**

Introduction :

- Decimation and interpolation are the two basic building blocks in the multirate digital signal processing systems. As the linear canonical transform (LCT) has been shown to be a powerful tool for optics and signal processing, it is worthwhile and interesting to analyze the decimation and interpolation in the LCT domain. .

Prerequisite knowledge for Complete understanding and learning of Topic:

- Interpolation

Detailed content of the Lecture:

Interpolation by a factor I $\quad y(m) = x(m/I)$ for $m = 0, \pm I, +2I, \ldots$

$$x(n) \longrightarrow v(m) = \begin{cases} x(m/I), & m = 0, \pm I, \pm 2I, \ldots \\ 0, & \text{otherwise} \end{cases}$$

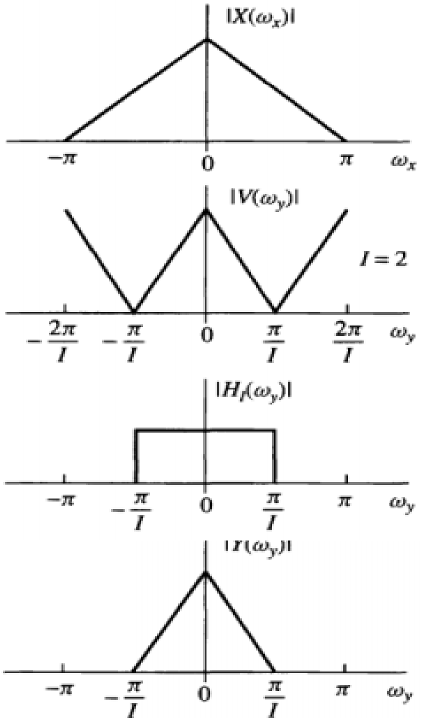
$$F_y = I F_x$$

$$V(z) = \sum_{m=-\infty}^{\infty} v(m) z^{-m} = \sum_{m=-\infty}^{\infty} x(m) z^{-mI} = X(z^I)$$

DTFT: $\quad V(\omega_y) = X(\omega_y I)$

$$\omega_y = 2\pi F/F_y \qquad F_y = I F_x \quad \Rightarrow \quad \omega_y = \frac{\omega_x}{I}$$

Interpolation by a factor I



As the frequency component of $x(n)$ are unique in the range $0 \le \omega_y \le \pi/I$ Images beyond that in $v(n)$ should be rejected by low pass filtering

$$H_I(\omega_y) = \begin{cases} C, & 0 \le |\omega_y| \le \pi/I \\ 0, & \text{otherwise} \end{cases}$$

$$y(m) = \sum_{k=-\infty}^{\infty} h(m-k)v(k)$$

$$Y(\omega_y) = \begin{cases} CX(\omega_y I), & 0 \le |\omega_y| \le \pi/I \\ 0, & \text{otherwise} \end{cases}$$

$$y(m) = \sum_{k=-\infty}^{\infty} h(m-kI)x(k)$$

C = ?

Interpolation by a factor I

$$y(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} Y(\omega_y)d\omega_y = \frac{C}{2\pi}\int_{-\pi/I}^{\pi/I} X(\omega_y I)d\omega_y$$

$$\omega_y = \omega_x/I, \longrightarrow = \frac{C}{I}\frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega_x)d\omega_x = \frac{C}{I}x(0)$$

$C = I$ is the desired normalization factor

Important Books/Journals for further learning including the page nos.:

6. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(784-789)**

Course Teacher

Verified by HOD

MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

L-43

LECTURE HANDOUTS

ECE | II/IV

Course Name with Code : 19ECC09/ DIGITAL SIGNAL PROCESSING

Course Teacher : Dr.T.R.Ganesh Babu. Prof./ECE

Unit : MULTIRATE AND DIGITAL SIGNAL PROCESSORS

Date of Lecture :

Topic of Lecture: Adaptive Filters: Introduction, Applications of adaptive filtering to equalization

Introduction :

- An adaptive filter is a system with a linear filter that has a transfer function controlled by variable parameters and a means to adjust those parameters according to an optimization algorithm. Because of the complexity of the optimization algorithms, almost all adaptive filters are digital filters.
- Adaptive filters are required for some applications because some parameters of the desired processing operation (for instance, the locations of reflective surfaces in a reverberant space) are not known in advance or are changing. The closed loop adaptive filter uses feedback in the form of an error signal to refine its transfer function.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Ideal Filters
- Linear Filtering

Detailed content of the Lecture:

- There are two input signals to the adaptive filter: and which are sometimes called the primary input and the reference input respectively.
- The adaption algorithm attempts to filter the reference input into a replica of the desired input by minimizing the residual signal, When the adaption is successful, the output of the filter is effectively an estimate of the desired signal.
- which includes the desired signal plus undesired interference and are correlated to some of the undesired interference in k represents the discrete sample number. The filter is controlled by a set of L+1 coefficients or weights.

Tapped delay line FIR filter

- If the variable filter has a tapped delay line Finite Impulse Response (FIR) structure, then the impulse response is equal to the filter coefficients. The output of the filter is given by

$$Y_k = \sum w_{lk} x_{(k-l)}$$

Adaptive Linear Combiner

- The adaptive linear combiner (ALC) resembles the adaptive tapped delay line FIR filter except that there is no assumed relationship between the X values.

- If the X values were from the outputs of a tapped delay line, then the combination of tapped delay line and ALC would comprise an adaptive filter. However, the X values could be the values of an array of pixels. Or they could be the outputs of multiple tapped delay lines. The ALC finds use as an adaptive beam former for arrays of hydrophones or antennas.

$$Y_k = \sum W_{lk} X_{lk}$$

Convergence

- Here μ(convergence) controls how fast and how well the algorithm converges to the optimum filter coefficients. If μ is too large, the algorithm will not converge.
- If μ is too small the algorithm converges slowly and may not be able to track changing conditions. If μ is large but not too large to prevent convergence, the algorithm reaches steady state rapidly but continuously overshoots the optimum weight vector.
- Sometimes, μ is made large at first for rapid convergence and then decreased to minimize overshoot.

Nonlinear Adaptive Filters

- The goal of nonlinear filters is to overcome limitation of linear models. There are some commonly used approaches: Volterra LMS, Kernel adaptive filter, Spline Adaptive Filter and Urysohn Adaptive Filter. The general idea behind Volterra LMS and Kernel LMS is to replace data samples by different nonlinear algebraic expressions.

$$Y_i = \sum W_j X_{ij}$$

Applications of Adaptive Filtering

- Linear Predictor
- Inverse Modeling
- Echo Cancellation
- Foetal Monitoring
- Noise Cancellation Filters

Important Books/Journals for further learning including the page nos.:

7. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(797-806)**

Course Teacher

Verified by HOD

L-44

LECTURE HANDOUTS

ECE **II/IV**

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **MULTIRATE AND DIGITAL SIGNAL PROCESSORS**

Date of Lecture :

Topic of Lecture: Subband Coding

Introduction :

- Sub-band coding (SBC) is any form of transform coding that breaks a signal into a number of different frequency bands, typically by using a fast Fourier transform, and encodes each one independently. This decomposition is often the first step in data compression for audio and video signals.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Linear Prediction

Detailed content of the Lecture:

Encoding audio signals

- The simplest way to digitally encode audio signals is pulse-code modulation (PCM), which is used on audio CDs, DAT recordings, and so on.
- Digitization transforms continuous signals into discrete ones by sampling a signal's amplitude at uniform intervals and rounding to the nearest value representable with the available number of bits. This process is fundamentally inexact, and involves two errors: discretization error, from sampling at intervals, and quantization error, from rounding.
- The more bits used to represent each sample, the finer the granularity in the digital representation, and thus the smaller the quantization error. Such quantization errors may be thought of as a type of noise, because they are effectively the difference between the original source and its binary representation.
- With PCM, the audible effects of these errors can be mitigated with dither and by using enough bits to ensure that the noise is low enough to be masked either by the signal itself or by other sources of noise.
- A high quality signal is possible, but at the cost of a high bitrate (e.g., over 700 kbit/s for one channel of CD audio). In effect, many bits are wasted in encoding masked portions of the signal because PCM makes no assumptions about how the human ear hears.
- Coding techniques reduce bitrate by exploiting known characteristics of the auditory system. A classic method is nonlinear PCM, such as the μ-law algorithm. Small signals are digitized with finer granularity than are large ones; the effect is to add noise that is

proportional to the signal strength.

Basic principles

- The utility of SBC is perhaps best illustrated with a specific example. When used for audio compression, SBC exploits auditory masking in the auditory system.
- Human ears are normally sensitive to a wide range of frequencies, but when a sufficiently loud signal is present at one frequency, the ear will not hear weaker signals.
- The basic idea of SBC is to enable a data reduction by discarding information about frequencies which are masked. The result differs from the original signal, but if the discarded information is chosen carefully, the difference will not be noticeable, or more importantly, objectionable.
- A digital filter bank divides the input signal spectrum into some number of subbands. The psychoacoustic model looks at the energy in each of these subbands, as well as in the original signal, and computes masking thresholds using psychoacoustic information.
- Each of the subband samples is quantized and encoded so as to keep the quantization noise below the dynamically computed masking threshold.
- The final step is to format all these quantized samples into groups of data called frames, to facilitate eventual playback by a decoder.
- Decoding is much easier than encoding, since no psychoacoustic model is involved. The frames are unpacked, subband samples are decoded, and a frequency-time mapping reconstructs an output audio signal.

Applications

- Beginning in the late 1980s, a standardization body, the Moving Picture Experts Group (MPEG), developed standards for coding of both audio and video. Subband coding resides at the heart of the popular MP3 format (more properly known as MPEG-1 Audio Layer III), for example.
- Sub-band coding is used in the G.722 codec which uses sub-band adaptive differential pulse code modulation (SB-ADPCM) within a bit rate of 64 kbit/s.
- In the SB-ADPCM technique, the frequency band is split into two sub-bands (higher and lower) and the signals in each sub-band are encoded using ADPCM.

Important Links/Website Details:
https://www.sciencedirect.com/topics/engineering/subband-coding

Important Books/Journals for further learning including the page nos.:

1. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(841-843)**

Course Teacher

Verified by HOD

L-45

LECTURE HANDOUTS

ECE **II/IV**

Course Name with Code : **19ECC09/ DIGITAL SIGNAL PROCESSING**

Course Teacher : **Dr.T.R.Ganesh Babu. Prof./ECE**

Unit : **MULTIRATE AND DIGITAL SIGNAL PROCESSORS**

Date of Lecture :

Topic of Lecture : Channel Vocoders

Introduction :

- Channel vocoder is a device for compressing, or encoding, the data needed to represent a speech waveform, while still retaining the intelligibilty of the original waveform.

Prerequisite knowledge for Complete understanding and learning of Topic:

- Sub Band Coding

Detailed content of the Lecture:

- The energy of each filter's output was then measured, or sampled, at regular time intervals and stored. This collection of filter energy samples then comprised the "coding" of the speech signal.
- This code could then be transmitted over a communication channel of lower bandwidth than would be neccessary for the raw speech signal. At the receiving end, the speech signal is reconstructed from this code by using the time sequence of filter energy samples to modulate the amplitude of a pulse signal being fed into a bank of filters similar to the ones used for the encoding.
- The channel vocoder, then, first analyzes the speech signal to estimate this time-varying spectral variation. To do this it uses the filters in the filter banks to determine how a particular frequency component of the speech signal is changing with time.
- On the output end, this analysis of the spectral variations are used to synthesize the speech signal by using another filter bank to apply these same spectral variations to an artificial periodic pulse like signal. The output filter bank acts as an artificial vocal track and the pulse signal acts as a set of artificial vocal chords.
- Some sounds produced during speech do not arise from the vocal chords, but are produced by turbulent air flow near constrictions in the vocal tract such as may occur between the tongue and the teeth. For example, such sounds as "SSS", "K", "SSHH", "P", and so forth arise in this manner.
- These sounds would be poorly reconstructed using a pulse excitation source, and so most channel vocoders also have a noise signal that can be used as an excitation source as well. A "Voiced/Unvoiced" detector circuit is used to detect whether the speech signal is arising from vocal chord excitation (Voiced speech) or is arising from noise excitation (Unvoiced

speech), and the appropriate excitation source is then selected at the output end.

- Channel vocoders were originally developed for signal coding purposes, with an eye (ear?) towards reducing the amount of data that would be needed to be transmitted over communication channels. In fact, speech coding system development continues to this day to be a vigorous area of research and development.
- These systems have far outstripped the basic channel vocoder idea in complexity, coding efficiency, and intelligibilty, however. So why do we still care about channel vocoders? The reason is that channel vocoders (and the functionally equivalent, but computationally quite different, phase vocoder) have found application to music production.

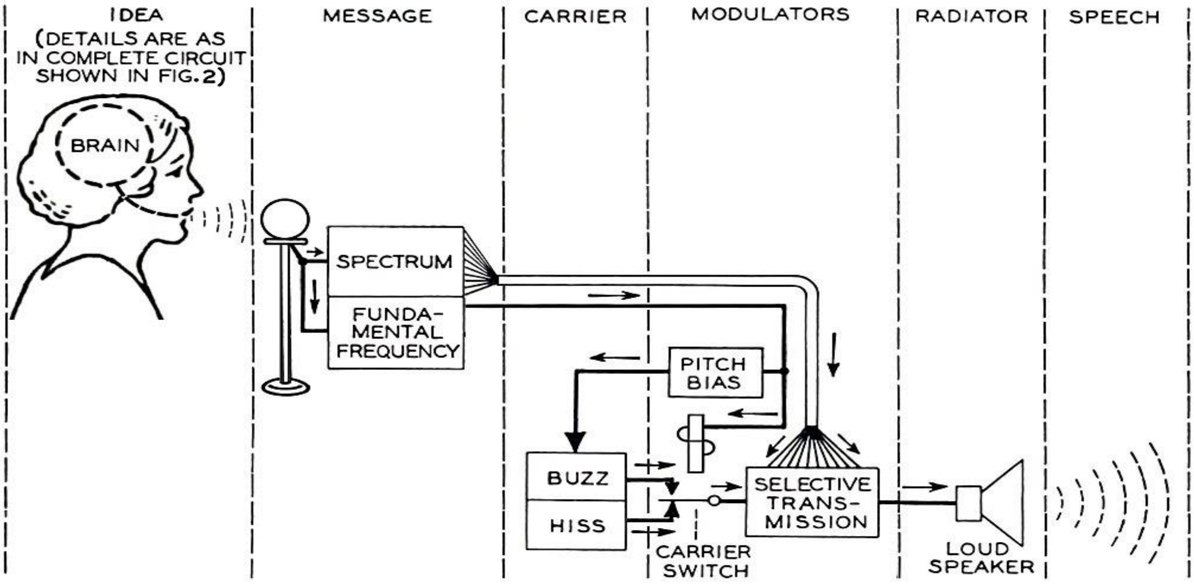


Fig. 7—Schematic circuit of the vocoder.

- The vowel sounds are pitched and have a definite spectral structure. The vocal filter module is designed to produce the spectrum of vowel sounds. Vowel sounds are created by passing a periodic pulse waveform (which sets the basic pitch of the voice) through a complex filter with multiple resonances.
- The pulse waveform models the vibration of the vocal cords. In the patches below we use a narrow pulse wave, but half-wave rectified sine-waves and sawtooth waves could also be used.
- The complex filter models the effect of the vocal cavity, formed by the mouth, throat, and nasal passages. Because of the complicated shape of these passages, some frequencies are enhanced while others are diminished in strength. This results in resonances and anti-resonances.

Important Links/Website Details:
https://www.cim.mcgill.ca/~clark/nordmodularbook/nm_speech.html

Important Books/Journals for further learning including the page nos.:

2. Digital Signal Processing Principles Algorithms & Applications, John G. Proakis & Dimitris G.Manolakis, 4th Edition. **Pg.No(841-845)**

Course Teacher

Verified by HOD